SANDIA REPORT

SAND2014-18981

Unlimited Release

Printed Month and Year

Using High Performance Computing to Examine the Processes of Neurogenesis Underlying Pattern Separation and Completion of Episodic Information

James B Aimone, Michael L Bernard, Steve Verzi, Craig M Vineyard

Prepared by
Sandia National Laboratories
Albuquerque, New Mexico 87185 and Livermore, California 94550

Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

Sandia National Laboratories

Issued by Sandia National Laboratories, operated for the United States Department of Energy by Sandia Corporation.

NOTICE: This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government, nor any agency thereof, nor any of their employees, nor any of their contractors, subcontractors, or their employees, make any warranty, express or implied, or assume any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represent that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government, any agency thereof, or any of their contractors or subcontractors. The views and opinions expressed herein do not necessarily state or reflect those of the United States Government, any agency thereof, or any of their contractors.

Printed in the United States of America. This report has been reproduced directly from the best available copy.

Available to DOE and DOE contractors from

U.S. Department of Energy
Office of Scientific and Technical Information
P.O. Box 62
Oak Ridge, TN 37831
Telephone: (865) 576-8401
Facsimile: (865) 576-5728
E-Mail: reports@adonis.osti.gov
Online ordering: http://www.osti.gov/bridge

Available to the public from

U.S. Department of Commerce
National Technical Information Service
5285 Port Royal Rd.
Springfield, VA 22161
Telephone: (800) 553-6847
Facsimile: (703) 605-6900
E-Mail: orders@ntis.fedworld.gov
Online order: http://www.ntis.gov/help/ordermethods.asp?loc=7-4-0#online

SAND2014-18981
Unlimited Release
Printed Month Year

Using High Performance Computing to Examine the Processes of Neurogenesis Underlying Pattern Separation and Completion of Episodic Information

James B Aimone[1], Michael L Bernard[1], Steve Verzi[2], and Craig M Vineyard[1]

[1]Cognitive Modeling Group

[2]Systems Research, Analysis, and Applications Group
Sandia National Laboratories
P.O. Box 5800
Albuquerque, New Mexico 87185-MS1327

Abstract

Adult neurogenesis in the hippocampus region of the brain is a neurobiological process that is believed to contribute to the brain's advanced abilities in complex pattern recognition and cognition. Here, we describe how realistic scale simulations of the neurogenesis process can offer both a unique perspective on the biological relevance of this process and confer computational insights that are suggestive of novel machine learning techniques. First, supercomputer based scaling studies of the neurogenesis process demonstrate how a small fraction of adult-born neurons have a uniquely larger impact in biologically realistic scaled networks. Second, we describe a novel technical approach by which the information content of ensembles of neurons can be estimated. Finally, we illustrate several examples of broader algorithmic impact of neurogenesis, including both extending existing machine learning approaches and novel approaches for intelligent sensing.

ACKNOWLEDGMENTS

CONTENTS

FIGURES

TABLES

NOMENCLATURE

AI	Artificial Intelligence
ACh	Acetylcholine
ANN	Artificial Neural Networks
BC	Basket Cells
CA1	Cornu Ammonus Area 1
CA3	Cornu Ammonus Area 3
CCK	Cholecystikinin
DG	Dentate Gyrus
GC	Granule Cells
HICAP	Hilar Interneurons projecting onto the Commisural / Associative Pathway
HIPP	Hilar Interneurons projecting onto the Perforant Pathway
MC	Mossy Cells
MS	Medial Septum
PV	Parvalbumin
SNL	Sandia National Laboratories

1. INTRODUCTION

The leveraging of neuroscience in the development of novel approaches to computation has long been suggested as a potential alternative to conventional computing technologies. The rationales for the brain's potential inspiration for computing are numerous: the low energy consumption of the brain, the ability to make high consequence decisions in short time spans, the ability to perform cognitive operations in complex contexts, the formation and use of episodic-like memories, and the ability to rapidly extract features and make deep insights based on observations of diverse multi-sensory scenes are all aspects of the brain's functions that would be revolutionary if implemented in computing systems. However, despite this powerful inspiration and multiple billions of dollars annually invested in neuroscience research in the United States alone, there is little notable impact of these neuroscience concepts on computational algorithm and architecture research today.

The field of computational neuroscience is increasingly being recognized as a potential link between the mostly experimental biological research and the engineering of brain-inspired computing systems. Computational neuroscience is quite distinct from historical computer science research areas such as artificial intelligence (AI) and machine learning. These fields have leveraged historic concepts from psychology and neuroscience but have made little effort to utilize modern neuroscience findings in their research. Indeed, this lack of a clear link between neuroscience and domains such as artificial neural networks (ANNs) and AI is not widely appreciated outside of those fields, leading to both misconceptions about the biological relevance of these computing approaches and generally providing confusion about what neural-inspired computing should look like.

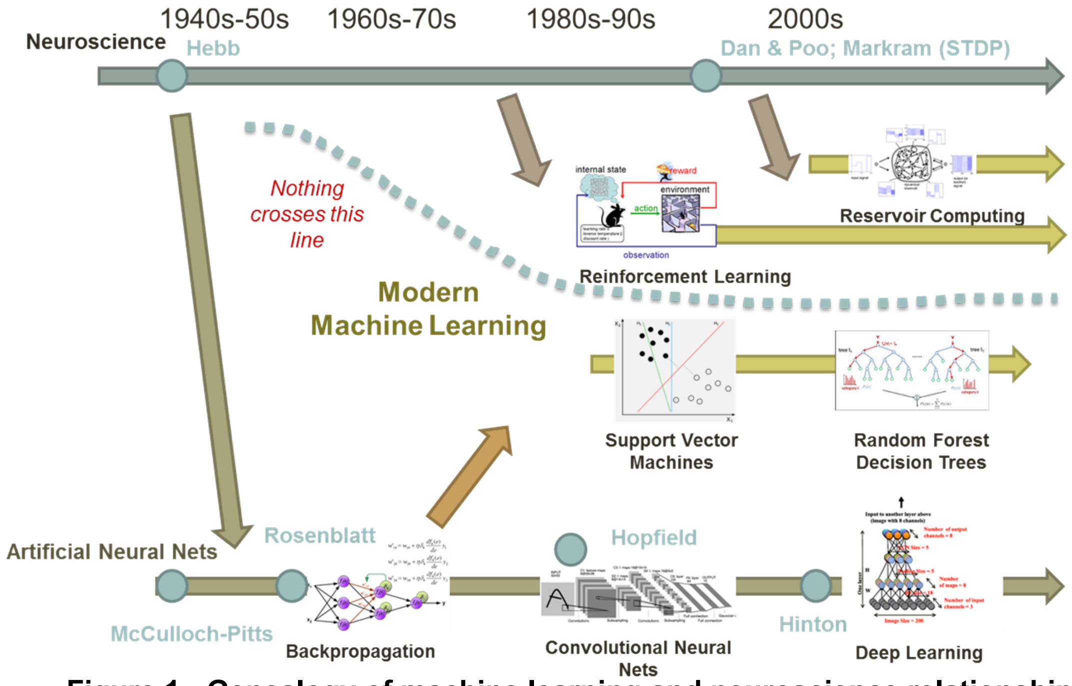


Figure 1 - Genealogy of machine learning and neuroscience relationship

The establishment of a clear strategy for leveraging neuroscience insights into computational methods (either existing or new) is potentially critical for fully realizing the aforementioned potential of brain-inspired computation as a post-Moore's Law technology base. Indeed, this technical path has been identified as a critical research direction in a number of locations, including the upcoming IARPA program MICRONS. Much of this research is related to the broader goals of President Obama's BRAIN Initiative, which has the goals of greatly increasing the resolution of systems neuroscience research and identifying paths to leverage that research in a number of application areas relevant to the BRAIN Initiative's sponsors, which include both health centric government agencies such as the National Institutes of Health and national security centric agencies such as the Department of Defense.

This report describes the efforts in the Episodic Memory / Neurogenesis LDRD which had goals very similar to the overall goals in the programs described above. Our research project focused on the computational neuroscience investigation of the adult neurogenesis process in the hippocampus, which is a process increasingly appreciated to be critical in higher cognitive function and episodic (event-based) memory formation. Furthermore, neurogenesis is a relatively recently appreciated and unique process in the brain which has had little influence outside of the neuroscience community. As a result, neurogenesis represents a neurobiological process that is both potentially of critical importance in achieving neural function in algorithms but is also absent from most conventional machine learning research.

The research performed in this project has impact in two areas. First, the neuroscience community has long lacked a strong theoretical and computational understanding of the value of neurogenesis in cognitive function. The modeling results in this study, along with the corresponding work in designing analytical techniques and validation approaches, represents a substantial contribution to the field. In particular, the scaling results described in chapter 3 are notable for their relevance to understanding neurogenesis function in humans. Second, the potential relevance of applying concepts based on neural dynamics and processes to machine learning algorithms has often been cited, but there remain few examples of effective translation. Chapter 5 will illustrate the potential long-term impact of this work in other application areas.

1.1. Biological process of neurogenesis

1.1.1. What is known

The study of the biological process of adult neurogenesis only began relatively recently; while observations of the neurogenesis process in the adult go back to the 1950s, it was only in the 1990s that extensive characterization and appreciation of the process occurred. Since then, the neuroscience community has rapidly characterized the neurogenesis process at all scales ranging from molecular and cellular studies to systems and behavioral levels. This research is extensive, and we reference the reader to the numerous reviews of this in literature, including several supported by this LDRD.

Briefly, the process of adult neurogenesis in the dentate gyrus (DG) region can be summarized as follows. Each day, roughly 1,000 new neurons are born from a stem cell population that resides locally within the DG. This rate is highly regulated by a number of intrinsic and extrinsic factors, as is the ultimate survival of the neurons that are born. While numbers differ somewhat from study to study, within several weeks about half of the neurons that are born no longer exist, most likely due to activation of apoptotic pathways (an internal gene signaling cellular death mechanism). If a neuron lives to about four to six weeks old, it most likely will persist indefinitely.

Notably, it is very possible – even likely – that there is a secondary cell death mechanism that impacts mature neurons. Most studies quantifying neurogenesis levels are in rats and mice, which live to roughly at most two years old. This relatively short lifespan makes it possible that a low neurogenesis rate (~5% a month surviving) that diminishes greatly with age (6 to 9 month old rats have markedly lower neurogenesis rates are considered old in neurogenesis studies) may not require a substantial cell death mechanism. However, the observation of neurogenesis in humans at roughly a 1-2% per year rate (which in absolute numbers is similar to levels in rodents) is suggestive that there must be a mechanism to replace embryonic and developmentally born neurons. Since quantification of cell death is challenging, this has remained a speculative notion.

In rodents, new neurons take approximately two months to achieve maturity. During this time, they progress from a "neuroblast" cell phenotype, which lacks the projections commonly associated with neurons, to fully functional granule cells that are indistinguishable from those born at earlier ages. This process requires the growth of a pronounced apical dendrite (the input projections to neurons) and an output axon called a "mossy fiber". By just over two weeks of age, these neurons begin developing *dendritic spines* which are the location of the input synapses onto the neuron's dendrites . Likewise, around this age they begin to form *axonal boutons* or new output synapses in the downstream hilus and CA3 regions.

Once new input and output synapses start to form at about 14 to 16 days old, the cells mature rapidly, obtaining new synapses at a rapid pace. By about two months old, the neurons have about 5,000 to 6,000 input glutamatergic synapses from both internal and external cortical populations. Glutamate is the major neurotransmitter of the brain, and it induces a positive current in the downstream neurons by opening a channel permeable to Na+ and K+ ions. Furthermore, mature DG neurons receive a significant number of inhibitory inputs from neurons referred to as interneurons. Interneurons release GABA as a neurotransmitter, which opens Cl- channels, thus lowering the voltage of the neuron further, which keeps it from firing. Notably, unlike ANNs, synapses in biological systems always use the same neurotransmitter. As a result, if a synapse is positive, it will always be positive; and vice versa for inhibitory synapses. A final key observation is that synapses on young neurons are more *plastic*, i.e., more amenable to learning, than those on mature neurons.

In addition to this difference in connectivity and synaptic plasticity, young neurons are distinct from the mature neurons in their basic electrophysiological properties. Young neurons typically have a higher membrane resistance, which allows individual synapses to have a higher relative impact than a similar weighted (same maximum conductance) synapse on a mature neuron. The combination of these properties – the physiology, connectivity, and plasticity of young neurons – has led to a widespread acceptance that these cells are more active or *hyperexcitable* compared to the mature population.

This increased excitability for young neurons is notable given the existing hypotheses for DG function. Prior to the widespread acknowledgement that the DG hosted new neurons, theorists and computational neuroscientists had generally come to a consensus that the DG was responsible for two key functions, both related to a key memory encoding role. Since the 1980s, the DG has been thought to be critical for driving the encoding of memories in the downstream

CA3 region . This “training” function was due in part to its sparse but powerful output synapses, which were potentially capable of individually driving a downstream CA3 pyramidal neuron. As a result, the DG could act like a quasi-supervised trainer to the rest of the hippocampus’s memory formation. The question has remained what the DG is training. One hypothesis is that the DG is simply responsible for selecting as orthogonal a sparse representation in the downstream CA3 network as possible to minimize interference between memories. At the extreme, this is effectively a “hash coding” hypothesis; each memory gets randomly assigned a subset of neurons in the DG, which then results in a random CA3 ensemble. In the high dimensions that are in play in the DG and hippocampus, a somewhat sparse activation could yield effectively unlimited number of combinations and a very low potential rate of interference. The other hypothesis is more information based; the DG is not producing an information-free hash code, but rather a very sparse code that is a function of the cortical inputs that drive the whole hippocampus. In this view, the information content of the DG can be extremely high, but the outcome is the same; a nearly orthogonal representation ensemble in the CA3 to encode whatever events are occurring.

The difference between these two hypotheses is subtle, and indeed it has caused a significant amount of debate in the broader hippocampus field (see http://www.patternseparation.com for active discussions on this topic). Notably, the implications for neurogenesis are potentially quite substantial. The bulk of neurogenesis behavioral studies, which are explained in detail in Aimone et al., 2014, have interpreted their results using the pattern separation hypothesis, however it is quite possible that alternative interpretation based on information coding may yet prove informative . It is in this context that the computational modeling work described here was motivated.

1.1.2. Open questions in neurogenesis research

Importantly, there remain a number of key questions outstanding in the neurogenesis community. Many of these reside at the cellular level, such as noting the genetic instructions that control the neurogenesis process and understanding which components regulate the proliferation and differentiation of these cells. While these questions are important, particularly for health related applications, their focus is too low level to have significant impact on the application focused research here.

What could have impact on the potential algorithmic impact of neurogenesis are current studies focusing on the functional impact of neurogenesis on the overall DG and hippocampus function. While the theories of neurogenesis function mentioned are well grounded in what is currently known about the process, it is possible, and even likely, that there are complexities about the interaction of young (or mature) neurons with other neurons in the DG that are currently not incorporated in anyone’s models or theories. This is a common challenge in neuroscience broadly; the overall complexity of the system is such that one is never entirely confident that all models and parameters are perfectly constrained. Nonetheless, it is worth noting that from a biological point of view, a number of current studies in the broader research community may at one point in the future identify system-level interactions of functions that to date are entirely unappreciated in the field.

1.2. Algorithmic leveraging of neurogenesis

1.2.1. Neurogenesis-inspired artificial neural networks

Historically, most artificial neural networks (ANNs) such as multi-layer perceptrons and other related network such as deep learning and recurrent neural networks have not used any structural dynamics in their generation or operation; rather most ANNs use a pre-constructed network and simply fit a set of synaptic weights to whatever data is available for training. In most applications, this pre-operation training has been sufficient; however it has been recognized in a number of areas that these ANNs are not actually well suited for online learning, which is essentially training the network to new data once trained on other data. Generally, these algorithms suffer from a phenomenon known as the *stability-plasticity dilemma,* whereby a network can either retain previously trained information or learn new information at the risk of losing the old information. Further, in the case of simple ANNs, many models trained through backpropagation, which is the standard training technique, are squarely on the "stability" side of that dilemma; empirically backpropagation often sends a model's weights into such a highly trained state that new data is not well suited for changing the weights to new conditions. As a result, heavily trained or overtrained ANNs often are incapable of learning new data well at all without other manipulations. In alternative configurations, ANNs that are forced to learn the new data may suffer from what is known as *catastrophic interference*, whereby the attempt to introduce a new memory into an existing ANN can collapse the entire attractor landscape of the network, leading too much of the information stored being lost.

Along these lines, there have been a few examples of ANNs incorporating a neurogenesis-like process to get past the limitations of ANNs in acquiring new information. These algorithms were initially quite basic, such as taking simple multi-layer perceptrons and training them using backpropagation to recognize one alphabet, and then demonstrating that the replacement of nodes in the hidden layer (the neurogenesis process) allows the network to learn a new alphabet (Figure 2). In this example, based on the 2004 Chambers study, a network capable of representing the Roman alphabet (left) is incapable of representing certain letters of the Greek alphabet, such as θ (middle), which are not directly corresponding to their Roman counterpart. Even with further online training, these networks often cannot learn the new alphabet. However, if the trained network has a marginal replacement of a number of hidden nodes (right), the network is then capable of learning with high fidelity the new alphabet.

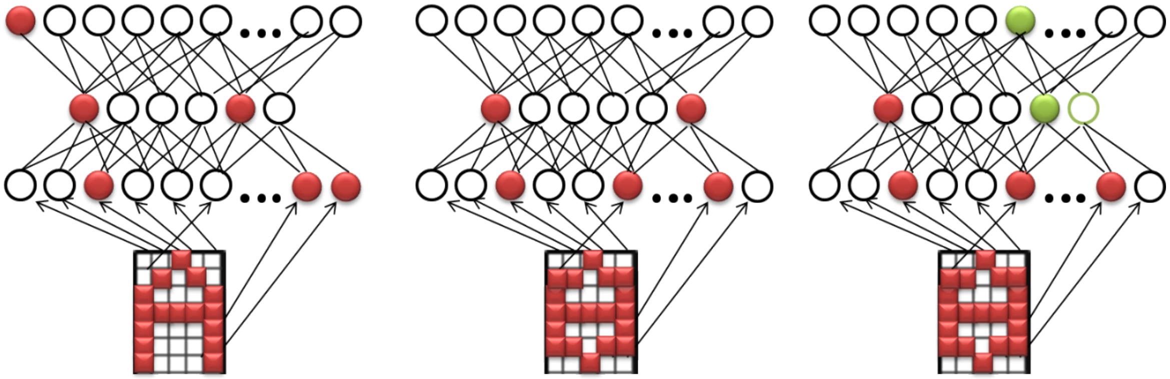


Figure 2 - Illustration of neurogenesis impact on simple ANNs

From this simple example, there were a number of more elaborate multi-layer perceptron studies, using a number of different training techniques. Within the basic ANN framework, methods using neural addition (as opposed to replacement), unsupervised Hebbian learning (as opposed to backpropagation), and autoencoder strategies (as opposed to classification networks) were used, each showing a similar benefit for neurogenesis in allowing an already trained network to learn new information.

1.2.2. Potential broader impact of neurogenesis in application directed models

The ANN-neurogenesis research illustrated above is indicative of a potential role of new neurons in making ANNs more robust to dynamic data acquisition, but they are nonetheless limited to the same limitations of ANNs in general. First, while the neurogenesis process in biology is quite dynamic (see section 1.1.1), ANNs rarely have time directly incorporated within their nodes' behavior, with generally only recurrent networks which evolve over a number of cycles having an ability to represent something akin to time. The introduction of new neurons gradually into a network likely provides a distinct temporal scale onto the system, but this has not been extensively explored. Second, while ANNs are theoretically a general structure for a number of node-connection based model architectures; in practice they implement a connectivity structure far different than anything seen in biology. ANNs typically are highly connected, with most (or every) neurons connecting to most (or every) other neurons in a neighboring layer. The differentiation between neurons thus only exists in the set of weights which are established during training. In contrast, the biological network that neurogenesis occurs in is highly structured, with a very distinct connectivity pattern and a number of distinct cell types that uniquely affect the dynamics and maturation of these new neurons. Finally, in the ANN condition, all nodes are effectively identical. In contrast, the neurogenesis process appears to be special precisely because the new neurons, or new nodes, are behaviorally and connectively distinct from those neurons in the existing population.

2. HIGH FIDELITY MODELING OF ADULT NEUROGENESIS

Our approach to computationally addressed both whether young neurons are contributing to pattern separation or an alternative function and whether new neurons are sufficiently valuable in humans. This work leverages insights from our previous model of DG neurogenesis, but uses a considerably higher temporal and anatomical resolution to assess the implications of new neurons on DG neuronal representations. Briefly, our model focuses on the DG region of the hippocampus along with its two primary inputs, the lateral (lEC) and medial (mEC) entorhinal cortices. The DG consisted of 7 layers, the principal neurogenic granule cell (GC) layer and six feedback and feed-forward interneuron layers, and the lEC and mEC were controlled to represent object and spatial grid cell information respectively . To model neuronal dynamics, we use Izhikevich spiking neurons, which is a reduced order model that is well suited for large scale simulations . Synapses were modeled using an exponential decay with an online spike-timing dependent plasticity (STDP) rule . Neurogenesis was implemented by young neurons gradually updating their dynamics and increasing their synaptic connectivity over time , and the model parameters of different neuron types were selected in accordance to physiology observations taken from several laboratories .

2.1. Model architecture

The neural network model was designed to capture the major neuron types and diverse interneuron classes in the dentate gyrus. While not every identified population of neurons was included, an effort was made to include most of the diversity present in the system. The network contained 9 full neuron layers (Table 1), several modulatory regions modeled as single neurons, and over 40 different connection types (Table 2).

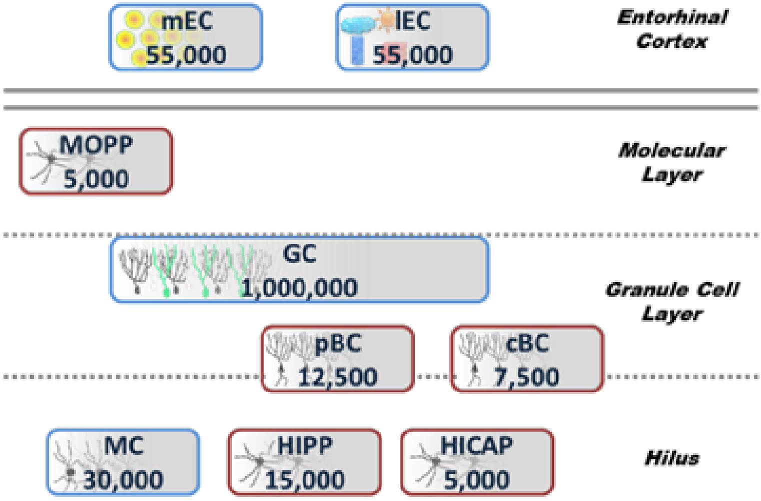


Figure 3 - Cell populations included in high fidelity simulations. Cell numbers are from rat.

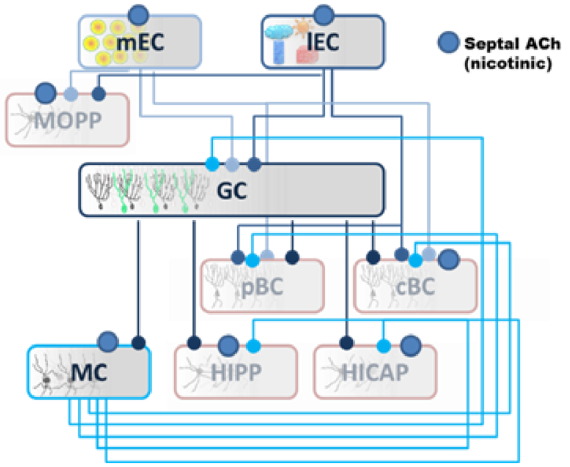


Figure 4 - Excitatory connections of high fidelity model

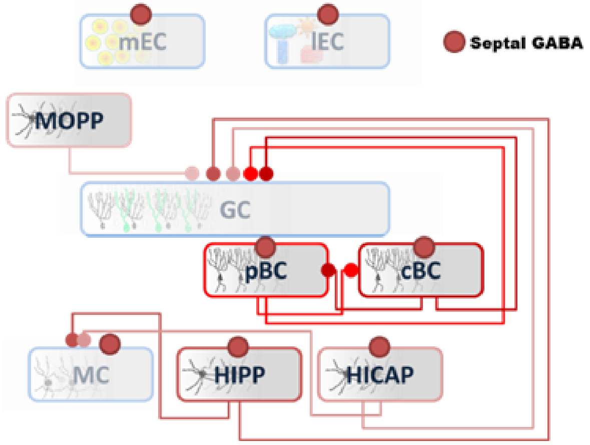


Figure 5 - Inhibitory connections of high fidelity model

Unlike our previous study (in which the model architecture emerged from a developmental process), the majority of the model was generated at the onset of the simulation in a fully developed form, and was static throughout its duration or simply had a small neurogenic population continually maturing over several days.

2.1.1. Neuron parameters and variables

At model initialization, each neuron was assigned a series of parameter values that were fixed for the course of the simulation. Some of these values were assigned based on their neuron type, and others were computed to fulfill another

The Izhikevich model used for neuron simulation is designed to approximate neuronal dynamics with more realism than an integrate and fire (or other simple) neuron, while more computationally tractable than a conductance based (e.g., Hodgkin-Huxley) model. The key parameters (a,b,c,d,k) are abstract with no direct biophysical link, although their effects can be correlated with different neuronal channels. In theory, these parameters can be selected to fit any neuron’s physiology; however in practice the fitting of the model to different neuron types is non-trivial and in this study the parameters were only loosely fit so as to approximate the general behavior of neurons observed in slice physiology studies.

Neurons were assigned a spatial location, either based on index (septo-temporal position) or randomly (transverse position and depth within layer). The x-location of each neuron (χ_x) was assigned sequentially ($\chi_x=n/N$). The y-location (χ_y) and z-location (χ_z) were assigned from a uniform random distribution between zero and one.

Neuron dynamics were computed according to a modified version of the model described by Izhikevich. Under the Izhikevich scheme, each neuron requires both parameters that explicitly represent biophysical features (capacitance, voltage threshold) and model-specific parameters that are selected to best fit the observed neuronal dynamics.

Biophysical Izhikevich parameters

C –capacitance of the neuron's soma

$E_{glutamate}$ – the reversal potential of glutamatergic receptors (0mV)

E_{GABA}– the reversal potential of GABA-A receptors (corresponds to the reversal of Cl-)

v_r – the resting potential of the neuron

v_t – the threshold potential of the neuron

I_{rheo} – the current typically required for the neuron to reach the threshold voltageif currently at resting potential

Model-specific Izhikevich parameters

a –Roughly corresponds to a decay constant of K+ channels

b – Roughly corresponds to weighting of voltage dependent K+ channels

c – The reset voltage of the neuron after an action potential

d – The spike-dependent reset of the '*u*' parameter. Roughly corresponds to the action potential activation of K+ channels

k – Sensitivity of the neuron to voltage deflections. Affects speed of action potential. Roughly corresponds to balance between Na+ and K+ conductances

k_1 - Parameter that increases 'k' during an action potential. Corresponds to voltage dependent Na+ channels

Neuron dynamic variables

v –Izhikevich parameter corresponding to a neuron's voltage.

u –Izhikevich "reset" parameter. Roughly corresponds to a neuron's open K+ conductance

2.1.2. Synapse parameters and variables

Synapse mapping

source – the pre-synaptic neuron ID. This neuron's activation determines whether the synapse may release neurotransmitter *target* – the post-synaptic neuron ID. This neuron will be the recipient of the synapses neurotransmitter release

Each synapse has a source neuron and a target neuron

type – Identification of the type of synapse. Many parameters are constant across type

dist – a measure of the time between the pre-synaptic and post-synaptic neuron

g_act – the absolute value of conductance achieved by activating the synapse

numsites – the number of release sites at the synapse (each has an independent probability of releasing transmitter)

Prob – the probability a pre-synaptic action potential will result in the release of transmitter

Synapse dynamic variables

g_tot – the realized value of conductance of the synapse

I_syn – the total current flow through the synapse

2.1.3. Generation of Connectivity

Synapse numbers in table X are based on the *target neuron*.

Synapse number scaling

$$N_{syn} = N_{syn,0} \times {}^{N_{source}}\!\big/_{N_{source,0}} \tag{1}$$

Each target neuron has a physical position in the network (x_t,y_t,z_t) and these positions are used to determine which source neurons it may receive inputs from. Each neuron in the source layer has a position (x_s, y_s, z_s) as well, and each synapse type has a variance (axonal/dendritic spread) associated with it in each dimension (x_{sig}, y_{sig}, z_{sig}). Notably, these dimensions are not all relevant for all connection types.

Each target neuron computes its probabilities of receiving an input from each neuron in the source layer according to normal distribution probability density functions that are based on relative positioning and connection type

Standard projections (only septo-temporal axis matters)

$$P_{syn}(i,j) = \mathrm{pdf}(\chi_{xi},\chi_{xj},\sigma_x) = \frac{1}{\sigma_x\sqrt{2\pi}}\exp\left(\frac{-(\chi_{xi}-\chi_{xj})^2}{2\sigma_x^2}\right) \tag{2}$$

Where χ_{xi} is the x-position of the ‘i' neuron, and χ_{xj} is the position of the ‘j’ neuron, and σ_x is the standard deviation of the x-positional spread.

Once these unscaled connection probabilities are computed, a normalized cumulative distribution (norm-cdf) is computed for each specific target neuron across all source neurons. For each of the number of desired source synapses, a random number is drawn from a uniform distribution between (0, 1) which is then compared to the cumulative distribution. For example, if the random number drawn is 0.5, then the source neuron situated where the norm-cdf passes 0.5 will be selected to form a synapse.

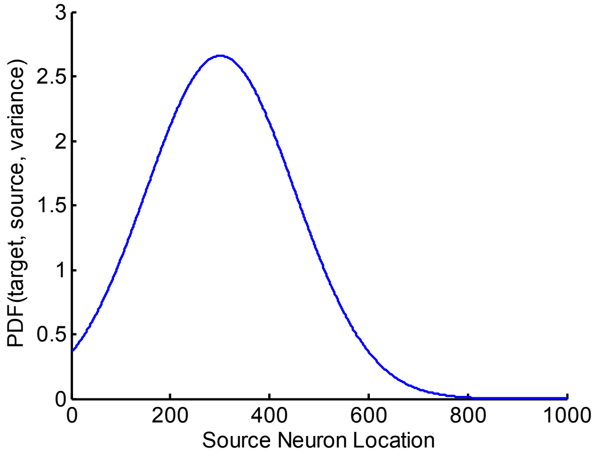


Figure 6 - Example probability density function of standard projections

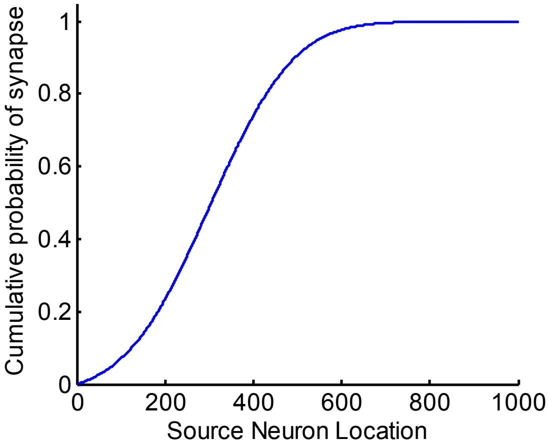


Figure 7 - Example cumulative density function of standard projections

Anti-topographical projection (mossy cell preferential avoidance of local S-T domain)

$$P_{syn}(i,j) = pdf(\chi_{yi},\chi_{yj},\sigma_y) \times pdf(\chi_{xi},\chi_{xj},\sigma_x) \times \left(pdf(\chi_{xi},\chi_{xj},\sigma_x) - pdf(\chi_{xi},\chi_{xj},0.5 \times \sigma_x)\right) \quad (3)$$

Note: Mossy cell axons preferentially avoid terminating on GCs (and presumably other DG neurons) in the local septo-temporal region. By subtracting a probability with a tighter standard deviation, neurons that are close together will have a "negative" probability of connecting

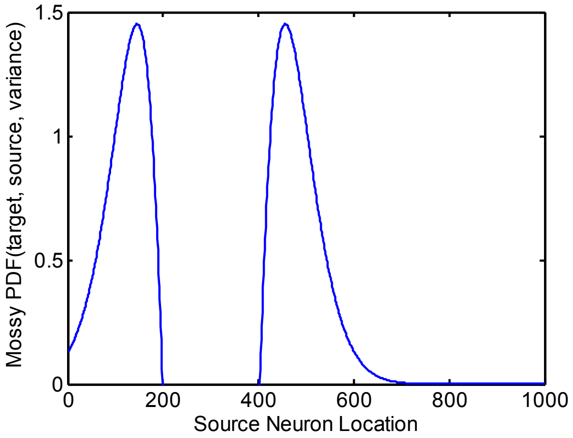


Figure 8 - Example probability density function of mossy cells

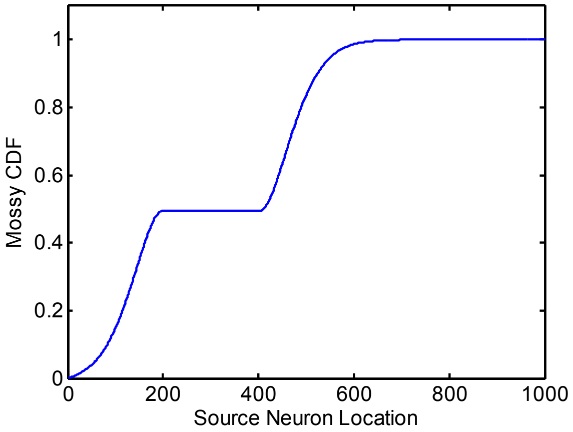


Figure 9 - Example cumulative density function of mossy cells

Notably, this algorithm guarantees that each target neuron receives a fixed number of synapses; however it does not ensure that source neurons will be equal in their impact. In fact, it is expected that the impact of source neurons on a downstream population will vary considerably (which is supported by biologically observations). Further, it is important to note that there is no check to keep neurons from receiving multiple inputs from the same source neuron. As a result it is possible, and indeed even common, for a neuron to have several source neurons from which it receives multiple synapses. Importantly, these synapses are uncoupled from each other aside from the common target; their weights, learning, and release probabilities are independently assessed.

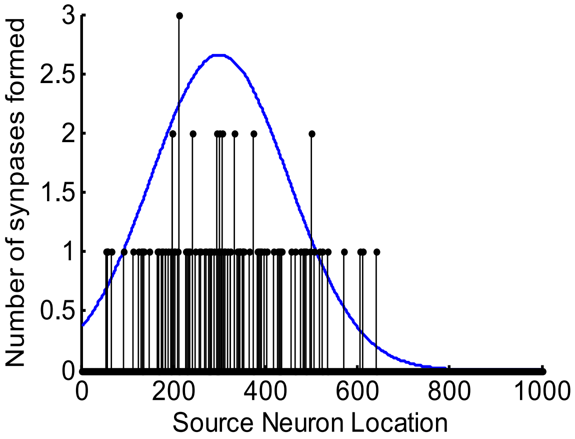


Figure 10 - Example synapse formation from standard projection

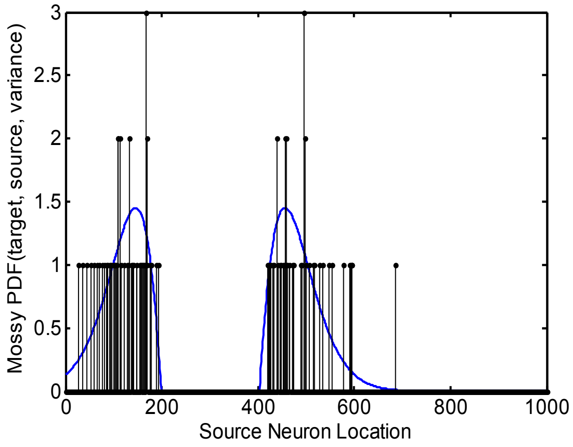


Figure 11 - Example synapse formation from mossy cell projection

A uniformly distributed random number is then generated for each possible source – target (i,j) projection in the mapping, and this number is then multiplied by its associated probability. Connections are subsequently generated for the N_{syn} highest values onto each target neuron.

Initializing connection strengths

Synaptic strengths were randomly selected using a log-normal distribution

$$\mu_{\mathrm{strength}} = \frac{N_{\mathrm{syn}}}{N_{\mathrm{syn,0}}} \times \mathrm{Conn}_{\mathrm{strength}} \tag{4}$$

$$\sigma_{\mathrm{strength}} = \mathrm{Conn}_{\mathrm{strength}_{\mathrm{var}}} \tag{5}$$

$$\sigma_{\mathrm{dist}} = \sqrt{\ln\left(\frac{\sigma_{\mathrm{strength}}}{\mu_{\mathrm{strength}}{}^{2}} + 1\right)} \tag{6}$$

$$\mu_{\mathrm{dist}} = \ln\left(\frac{\mu_{\mathrm{strength}}{}^{2}}{\sqrt{\mu_{\mathrm{strength}}{}^{2} + \sigma_{\mathrm{strength}}}}\right) \tag{7}$$

$$g_{act,ij} = \exp(\mu_{dist} + \sigma_{dist} \times \eta(0,1)) \tag{8}$$

Where $\eta(0,1)$ is a random number from a Gaussian distribution of mean 0 and standard deviation 1.

Furthermore, each synapse class was given a synaptic delay to represent axonal and dendritic conductance delays. These conductance delays were constant for a given synapse class, they did not vary for dorsal ventral locations.

2.1.4. Generation of immature neurons

At the onset of each simulation, a target percentage of neurons, $P_{young,}$ was identified. A random fraction of P_{young} GC neurons were classified as "young" and their age was determined by uniformly distributing those neurons identified as "young" across ages from -34 to 55 days old. Those neurons that were assigned a negative age represent those yet to be "born".

The spiking ability of GCs changes dramatically as they progress from 19 days old until about 25 days old . This maturation dynamic was implemented by altering the relevant Izhikevich parameters for neurons within this age range. Each age had a set of distinct Izhikevich parameters that were used to define its dynamics.

Similarly, the density of synaptic inputs changed according to this sigmoidal function

$$p_{age} = 0.5 \times \left(1 + \tanh\left(\frac{age - \mu_p}{\sigma_p}\right)\right) \tag{9}$$

Synapses were then retained for neurons below 56 days old according to this probability. Neurons below 0 days old (effectively those that are not born yet) could not have any synapses.

In addition, **neurons below 20 days old were not allowed to communicate their spikes**. While under some simulation conditions their dynamics and synaptic connectivity could potentially permit spiking, their spikes were neither recorded nor impactful on other neurons.

The final simulation results described here do not involve maturation over time, however the model was designed with this capability and the model was tested over multiple days. Maturation is implemented by advancing the age of each immature neuron by one day, which directly leads to re-setting of the neuron's dynamic parameters to those of its new age. Further, the gain of age may result in new synapses being formed, effectively based on the derivative of equation 12. Each new synapse that is formed is generated according to the rules outlined above.

2.2. Model dynamics

2.2.1. Neuron dynamics

Active neuron dynamics in the soma were simulated using Izhikevich dynamics. The Izhikevich neuron model uses two state variables - '*v*', which represents voltage (mV), and '*u*', which is an inactivation variable. Below an action potential peak (v_{peak} ***= 35mV***), '*v*' and '*u*' are updated according to the following equations.

$$\frac{dv}{dt} = \frac{1}{C}\left(k(v) \times (v - v_r) \times (v - v_t) - u + I_{syn} + I_{comp}\right)$$
$$\text{where...}$$
$$k(v) = k_1 + k_2 \times \tanh(v - v_t) \tag{10}$$

$$\frac{du}{dt} = a \times \left(b \times (v - v_r) - u\right) \tag{11}$$

$$\begin{aligned} &\text{if } v > 35 \\ &u \leftarrow u + d \\ &v \leftarrow c \end{aligned} \tag{12}$$

$$\begin{aligned} &\text{if } v < E_{GABA} \\ &v \leftarrow E_{GABA} \end{aligned} \tag{13}$$

When the neuron reaches threshold ($v>v_{peak}$), the neuron is considered to have fired an action potential, and is reset to a reset voltage ('*c*' parameter) and the 'u' variable is reset to a separate value ('*d*' parameter).

Table 1 lists the Izhikevich parameters used for the different neurons in the model. These parameters were selected manually to approximate the behavior of DG neurons in slice.

2.2.2. Synapse dynamics

Synapses were simulated with a simple time-decay conductance rule. At the onset of a synaptic event, the conductance was simulated using an instantaneous rise time, followed a simple decay with a time constant 'τ_{syn}'. At each time step, the driving forces of the open channels were updated.

To determine if a synapse receives an action potential (and thus is capable of releasing transmitter) at time 't', the synapse referenced whether its source neuron fired an action potential at time '$t\text{-}t_{dist}$'. For each synapse

$$syn_{vesicles} = \delta\left(f_{source}(t - t_{dist}) - 1\right) \times B(N_{sites}, p_{syn}) \tag{14}$$

Where *δ(•)* is a delta function determining whether the source neuron fired at time $t\text{-}t_{dist}$, and *B(•)* is a random number selected from the binomial distribution with parameters corresponding to the number of synaptic release sites and the probability each site is activated upon receiving an action potential.

Synapse equations

At each time step, every synapse is updated according to decay from its previous state and addition of new synaptic events

$$\frac{dg_{syn}}{dt} = -\frac{1}{\tau_{syn}} \times g_{syn} \tag{15}$$

Synaptic events resulted in the potential release of neurotransmitter at the synapse, but this release is stochastic according to the synapses' probabilities of release.

If transmitter released

$$g_{syn} \leftarrow g_{syn} + g_{active} \times syn_{vesicles} \tag{16}$$

Total synaptic input currents are then summed for each neuron, based on the conductances and driving forces of the active synapses:

$$I_{syn,i} = g_{syn,i} \times (E_{trans,i} - v) \tag{17}$$

$$I_{syn} = \sum_{i = \text{input syns}} I_{syn,i} \tag{18}$$

Table 2 lists the synaptic strengths, neurotransmitters (and reversals) and time constants for each synapse type.

2.2.3. Synaptic Plasticity

Synaptic plasticity rules were taken from Clopath and Gerstners' online spike-timing dependent plasticity (STDP) model . This STDP model has a number of tunable parameters, which is capable of showing a number of different learning profiles. We instantiated the learning under conditions which induced a negative weight change if a post-synaptic spike occurs *before* a synaptic event, and a positive weight change if a post-synaptic spike occurs *after* a synaptic event.

Because of the nature of the long-term depression (LTD), negative learning could only occur at a time step when a pre-synaptic spike arrives and the post-synaptic voltage is above a threshold.

$$\frac{d}{dt}w^{-} = -A_{LTD} \times X(t) \times \max(v_{neg}(t) - v_r, 0) \tag{19}$$

$$\tau_{neg}\frac{d}{dt}v_{neg} = v(t) - v_{neg}(t) \tag{20}$$

Where $v_{neg}(t)$ is a filtered postsynaptic voltage, τ_{neg} is the time constant of the postsynaptic voltage filtering, A_{LTD} is the negative learning rate, and $X(t)$ is a Boolean value of whether a pre-synaptic neuron spike arrived at time t.

In contrast, positive learning, or long-term potentiation (LTP), can occur when a post-synaptic event (usually a spike) occurs at any of a number of time steps after a pre-synaptic spike arrives, and as a result operates using a filtered spike train as its pre-synaptic input. Positive learning is governed by the following equations

$$\frac{d}{dt}w^{+} = A_{LTP} \times x_{filt}(t) \times \max(v(t) - v_t, 0) \times \max(v_{pos}(t) - v_r, 0) \tag{21}$$

$$\tau_{pos}\frac{d}{dt}v_{pos} = v(t) - v_{pos}(t) \tag{22}$$

$$\tau_x\frac{d}{dt}x_{filt} = X(t) - x_{filt}(t) \tag{23}$$

in which $v_{pos}(t)$ is a separate filtered postsynaptic voltage, with τ_{pos} as the time constant, A_{LTP} is the positive learning rate, $x_{filt}(t)$ is a filtered pre-synaptic spike train with τ_x as its time constant.

2.3. Simulation inputs and parameters

2.3.1. Input neuron firing distributions

There were four inputs to the model: a single medial septum GABA neuron, a single medial septum acetylcholine neuron, a population of projecting lEC neurons, and a population of projecting mEC neurons.

The MS-GABA septal neuron was simulated by providing a depolarization of I=120pA, which induced natural oscillations in its activity based on the tuned parameters. This represented the theta source in the network.

The MS-ACh neuron was given a constant depolarization of I=260pA, but was inhibited by the MS-GABA neuron.

EC neurons were given a random depolarization based on the context, objects and spatial location in which the model is situated. The contextual inputs and object-based inputs were based on a skewed distribution which was close to zero for most neurons and roughly uniformly distributed for the rest of neurons. The initial distribution of input currents onto input EC neurons were given by:

$$\text{global}(n) = \exp\square(-40 \times U(0,1)^2) \tag{24}$$

$$\text{object}(n,m) = \exp\square(-40 \times U(0,1)^2) \tag{25}$$

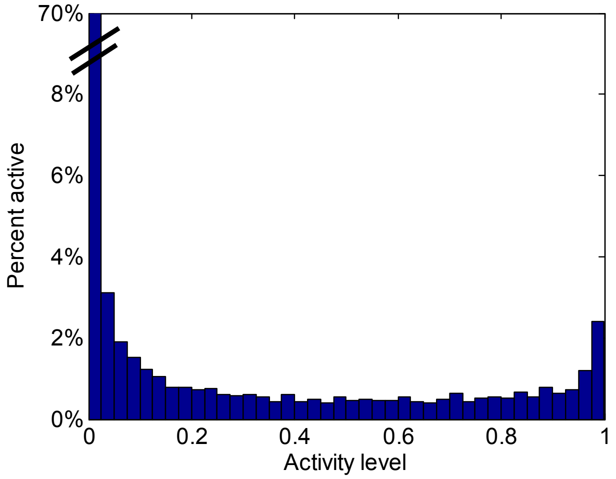


Figure 12 - Example activity level distribution of cortical inputs

Lateral EC neurons were object guided, as a result their activity was dependent on whatever object was being attended to (see below).

$$\text{lEC}(n) = \text{Irheo} \times \text{lECobj} \times \text{object}(n,m) \tag{26}$$

Or, if objects are not being used in the simulation or the lEC is not attending to objects:

$$\text{lEC}(n) = \text{Irheo} \times \text{lECglobal} \times \text{global}(n) \tag{27}$$

In contrast, medial EC neurons were a combination of contextual inputs and spatial grid cell contributions.

$$\text{mEC}(n) = \text{Irheo} \times \text{mECglobal} \times \text{global}(n) \times G(n,x,y) \tag{28}$$

During the simulation, these input activities are input to neurons as if they are synaptic inputs.

2.3.2. Object Selection

In any given environment, there are a set of objects that can potentially be “seen” by the model. In most of our simulations, these were constrained to a circular set of locations outside of the path taken by the model, however our approach to assessing which object is attended to at any given instant was not dependent on this.

Suppose (x,y) determined the position of the model at a given time, and (δx, δy) determined the direction of the model’s path. Let (x_a, y_a) represent the position of an object *a*, then the angle of movement θ_{dx}, the angular offset of the object θ_a, and the Euclidean distance to the object D_a, can be given by

$$\theta_{dx} = \mathrm{atan2}\Box(\delta x,\delta y) \tag{29}$$

$$\theta_a = \mathrm{atan2}\left(x - x_a, y - y_a\right) - \theta_{dx} \tag{30}$$

$$D_a = \sqrt{(x - x_a)^2 + (y - y_a)^2} \tag{31}$$

with both θ_{dx} and θ_a set to be between -π and π.

From these, a saliency bias metric, S, was generated for object a

$$S_a = {}^{1}\!\big/\!\left(D_a * \theta_a^{\,2}\right) \tag{32}$$

The structure of this metric greatly penalizes objects that are distant or peripheral to the direction of motion, and approaches infinity for those objects that are directly in the line of sight or very close physically.

This bias metric is computed for all objects in the system, and then each is normalized by the cumulative object saliency at that position and time. A uniform random number (0,1) is this drawn and mapped to a CDF of those values to select an object to attend to. Objects with high saliencies are more likely to be chosen.

2.3.3. Grid Cell Activation

The mEC neurons were given a similar random depolarization as the lEC neurons. However, as opposed to a constant depolarization, the mEC neurons were modulated by spatial location based on a grid cell behavior.

Prior to simulation, each mEC neuron was assigned three parameters: λ – grid cell size/frequency ($\lambda =3- \chi_x *1.5$), θ – grid cell rotation ($\theta=2\pi*rand(0,1)$), and (φ_x, φ_y) – spatial offsets for the grid (each from uniform distribution).

$$k1 = \frac{4\pi \times \lambda}{\sqrt{6}} \times \left(\left(\cos\left(\theta + \frac{\pi}{12}\right) + \sin\left(\theta + \frac{\pi}{12}\right)\right) \times (x - \varphi_x) + \left(\cos\left(\theta + \frac{\pi}{12}\right) - \sin\left(\theta + \frac{\pi}{12}\right)\right) \times (y - \varphi_y)\right) \tag{33}$$

$$k2 = \frac{4\pi \times \lambda}{\sqrt{6}} \times \left(\left(\cos\left(\theta + \frac{5\pi}{12}\right) + \sin\left(\theta + \frac{5\pi}{12}\right)\right) \times (x - \varphi_x) + \left(\cos\left(\theta + \frac{5\pi}{12}\right) - \sin\left(\theta + \frac{5\pi}{12}\right)\right) \times (y - \varphi_y)\right) \tag{34}$$

$$k3 = \frac{4\pi \times \lambda}{\sqrt{6}} \times \left(\left(\cos\left(\theta + \frac{3\pi}{4}\right) + \sin\left(\theta + \frac{3\pi}{4}\right)\right) \times (x - \varphi_x) + \left(\cos\left(\theta + \frac{3\pi}{4}\right) - \sin\left(\theta + \frac{3\pi}{4}\right)\right) \times (y - \varphi_y)\right) \tag{35}$$

$$G(n,x,y) = \frac{2}{3} \times \left(\frac{\cos(k1) + \cos(k2) + \cos\Box(k3)}{3} + 0.5\right) \tag{36}$$

Both the mEC and lEC populations were also inhibited by the MS-GABA input.

2.3.4. Simulations

Simulations consisted of providing constant inputs to lEC neurons to represent the current environment, while varying the spatial location to simulate a path around a 1m diameter circle to vary the mEC grid cell behavior. Inputs to the model are designed to examine the integration of spatial information, which is encoded by the mEC, with object information, which is provided by the lEC. On each simulation day, the model receives inputs corresponding to an animal moving around a circular track within a context while observing a selection of different objects.

This study focused on testing the network's response to two different contexts; the first context (the "familiar" context) was previously "pre-trained" in the simulation (see *Pre-training* below), and a novel context which consists of a distinct contextual input and different object-location pairings. Novel contexts can potentially share some objects with familiar contexts, though their spatial locations are different, and similarly mEC neurons are structured as grid cells in both, though individual neuron activities change.

Our primary study focused on simulations of the circuit at different scales ranging from 10,000 GCs to 10,000,000 GCs. As the scaling below biologically realistic sizes is arbitrary, we used a simple linear approach to scale down the number of other neuron populations and synapses, maintaining the overall statistics of connectivity. For example, since in a rat DG a typically GC has about 5,000 excitatory inputs and 1700 inhibitory inputs, a GC within 1-100 reduced model would have only 50 excitatory inputs and 17 inhibitory inputs, each having its maximum conductance increased by a factor of 100. Thus the ratio between excitatory and inhibitory conductances is held constant with scaling, as are the overall physiology parameters. Notably, for lack of strong biological quantification of synaptic numbers, the scaling above rat size to monkey and human scales does not involve increasing the numbers of synapses further. (Note: this assumption is intentionally conservative as our results suggest that higher synaptic densities would further amplify the results we describe in the main text).

While the results of this study focus on a single day, this paradigm was developed such that the model can progress through time while adding new contexts, with the model receiving inputs representing the very familiar context, a different familiar context, and a novel context on each day.

2.4. Implementation

2.4.1. Overview of implementation

The model was implemented in C++ with OpenMPI to enable parallelization (see Parallelization of Model section below). All simulations were run on either the small Linux cluster "Claustrum" which has 256 processors and 2TB RAM, or the Red Sky supercomputer, on which we used a variable number of nodes, with each node having up to 8 compute cores and 12GB RAM.

By far the limiting factor in the simulation is the demand of synapses on overall system memory. Synapses in the model were 16 bytes (B), with each synapses having representations of source neuron (4B integer), target neuron (4B integer), maximum conductance (4B floating point), relative conductance (1B integer), synapse type (1B integer), and immediate spike history (1B Boolean). A major savings was achieved by discretizing the decay of synapses. For each time constant, τ, we preallocated a set of relative temporal decays in memory (e.g., 13 time steps out with a 5ms decay is 0.0743), and thus each synapse simply needed to track how long it had been since its previous full activation began.

Neurons were 128 bytes each, with a number of floating point variables (including all the Izhikevich model parameters). While maintaining this number of floating points was not particularly efficient from a memory perspective, keeping these values local to each neuron sped up computation considerably. Further, since there were typically several orders of magnitude fewer neurons, there was little benefit in optimizing neuron size.

All calculations were performed in floating point as opposed to double precision, which both dramatically cut memory demands but also sped up simulation time by approximately 20-30%. In original calculations performed in double precision we saw no meaningful differences in qualitative behavior of the model.

2.4.2. Parallelization of Model

In order to simulate the model at sizes comparable to mouse DG sizes, it was necessary to distribute the model across many different processors (aka, nodes). Distributing large models across many nodes presents considerable computational advantages in dividing the necessary computations necessary for simulation and reducing the total active memory load required on any given system. The principal downside for distribution of a model across different nodes is the communication cost and the fact that the overall system can only go as fast as the slowest node. These considerations imply that a well-distributed model will minimize information that must be transferred between nodes as well as evenly parse out the computational duties such that each node has roughly an equivalent amount of work to perform, minimizing the time that nodes do not have to wait for the slowest node to finish.

Here, we distributed our model DG network over *Np* processors by uniformly distributing the individual neurons across the nodes in sequence (neuron 1 went to node 1, neuron 2 to node 2, etc). Because neuron IDs were sequenced based on cell population (neurons 1-100000 were lEC, 100001-200000 were mEC, etc), this resulted in roughly equivalent numbers of neurons from each cell group assigned to each node. Next, each neuron's afferent (input) synapses - but *not* efferent (output) synapses – was then generated on that same node.

This distribution scheme resulted in essentially the same complements of neurons and synapses allotted to each node. As a result, while different neurons behaved differently, the overall population on each node was similar. As models got larger, the relative differences between the nodes were reduced, resulting in comparable processing loads across nodes. This allowed wait times to be minimized in the network.

Communication demands were also low in this distribution design. At each synaptic time step (1 millisecond), each node simply needed to communicate a list of the active neurons. Since activity in the DG system is highly sparse (most neurons in the DG/EC network fire at less than 1Hz), few bytes of information needed to be communicated at each time step. Each node then kept track of the history of activity of *all* neurons in the network over time (this results in the same information stored on each node, but this information is highly sparse). When computing synaptic activity, each synapse needs only access this stored firing matrix rather than accessing other nodes.

Table 1 Neuron parameters

Name	Count	vt	vr	C	k	a	b	c	d	k1	Irheo	glob_mult	obj_mult	tau_n	tau_p
lEC	55000	-35	-70	100	0.2	0.04	-0.2	-65	100	10	45	2	3	--	--
mEC	55000	-35	-70	100	0.2	0.04	-0.2	-65	100	10	45	8	0	--	--
MOPP	5000	-35	-50	100	0.01	0.03	-0.2	-52	40	10	--	--	--	--	--
GC	1000000	-35	-75	120	0.075	0.001	1	-55	25	30	--	--	--	10	10
PV-BC	12500	-35	-62	100	0.4	0.1	0.3	-55	20	10	--	--	--	10	10
CCK-BC	7500	-35	-62	100	0.4	0.1	0.3	-55	20	10	--	--	--	10	10
MC	30000	-45	-65	60	0.125	0.025	2	-55	250	10	--	--	--	10	10
HIPP	15000	-35	-65	27.646	0.01	0.004	-0.1	-60	40	100	--	--	--	10	10
HICAP	5000	-30	-65	27.646	0.1	0.02	2	-30	75	10	--	--	--	10	10
MS-Ach	1	-40	-60	50	0.1	0.02	10	-30	75	10	--	--	--	30	20
MS-GABA	1	-40	-60	50	0.1	0.015	10	-30	40	10	--	--	--	30	20

Table 2 Connectivity parameters

Name	N_syn	E_rev	strength	strength variance	x_var	y_var	spread_type	distance_abs (ms)	tau (ms)	p_release	transmitter	release site	A_LTD	A_LTP	tau_pre
lEC to MOPP	2000	0	0.0286	1	0.1	Inf	1	13	2	0.9	Glu	1	0	0	10
lEC to GC	2000	0	0.0286	1	0.1	Inf	1	13	4	0.9	Glu	1	3e-5	1e-5	5
lEC to pBC	2000	0	0.0545	1	0.1	Inf	1	15	2	0.9	Glu	1	0	0	10
lEC to cBC	2000	0	0.0545	1	0.1	Inf	1	15	2	0.9	Glu	1	0	0	10
mEC to MOPP	2000	0	0.0286	1	0.1	Inf	1	12	2	0.9	Glu	1	0	0	10
mEC to GC	2000	0	0.0229	1	0.1	Inf	1	14	4	0.9	Glu	1	3e-5	1e-5	5
mEC to pBC	2000	0	0.0545	1	0.1	Inf	1	14	2	0.9	Glu	1	0	0	10
mEC to cBC	2000	0	0.0545	1	0.1	Inf	1	14	2	0.9	Glu	1	0	0	10
MOPP to GC	100	-80	0.15	0.3	0.1	0.3	1	2	4	1	GABA	1	0	0	10
GC to pBC	12000	0	0.0818	1	0.04	0.1	2	3	2	0.2	Glu	1	0	0	10
GC to cBC	3000	0	0.182	0.5	0.04	0.1	2	3	2	0.25	Glu	3	0	0	10
GC to MCprx	100	0	0.714	0.5	0.04	0.1	2	3	2	0.4	Glu	4	0	0	10
GC to MCdst	1000	0	0.0714	1	0.04	0.1	2	3	2	0.4	Glu	4	1e-5	1e-5	5
GC to HIPP	3500	0	0.269	0.5	0.04	0.1	2	3	2	0.4	Glu	2	0	0	10
GC to HICAP	3500	0	0.269	0.5	0.04	0.1	2	3	2	0.4	Glu	2	0	0	10

pBC to GC	300	-80	0.270	0.3	0.15	0.3	1	2	4	0.95	GABA	1	0	0	10
pBC to pBC	150	-80	0.287	0.3	0.15	0.3	1	2	4	1	GABA	1	0	0	10
cBC to GC	300	-80	0.270	0.3	0.15	0.3	1	2	4	0.95	GABA	1	0	0	10
cBC to cBC	150	-80	0.287	0.3	0.15	0.3	1	2	4	1	GABA	1	0	0	10
MC to GC	1000	0	0.0246	1	0.3	0.3	3	10	2	0.9	Glu	1	1e-5	1e-5	5
MC to pBC	1500	0	0.0545	1	0.15	0.3	3	10	2	0.9	Glu	1	0	0	10
MC to cBC	1500	0	0.0545	1	0.15	0.3	3	10	2	0.9	Glu	1	0	0	10
HIPP to GC	500	-80	0.13	0.3	0.08	0.2	1	3	4	1	GABA	1	0	0	10
HICAP to GC	500	-80	0.13	0.3	0.08	0.2	1	3	4	1	GABA	1	0	0	10
pBC to MC	300	-80	0.270	0.3	0.15	0.3	1	2	4	0.95	GABA	1	0	0	10
cBC to MC	300	-80	0.270	0.3	0.15	0.3	1	2	4	0.95	GABA	1	0	0	10
HIPP to MC	500	-80	0.07	0.3	0.08	0.2	1	2	4	1	GABA	1	0	0	10
HICAP to MC	500	-80	0.07	0.3	0.08	0.2	1	2	4	1	GABA	1	0	0	10
MC to HIPP	1500	0	0.0308	1	0.15	0.3	3	2	2	0.2	Glu	3	0	0	10
MC to HICAP	1500	0	0.0308	1	0.15	0.3	3	2	2	0.2	Glu	3	0	0	10
nACh to MOPP	1	0	1.7143	0	1	Inf	0	2	30	1	ACh	1	0	0	10
nACh to cBC	1	0	2.1818	0	1	Inf	0	2	30	1	ACh	1	0	0	10
nACh to MC	1	0	0.7143	0	1	Inf	0	2	30	1	ACh	2	0	0	10
nACh to HIPP	1	0	0.7143	0	1	Inf	0	2	30	1	ACh	1	0	0	10

nACh to HICAP	1	0	0.7143	0	1	Inf	0	2	30	1	ACh	1	0	0	10
5ht3 to cBC	1	0	1.6901	0	1	Inf	0	2	2	1	5HT	1	0	0	10
MS-GABA to lEC	1	-80	10	0	1	Inf	0	15	4	0.8	GABA	4	0	0	10
MS-GABA to mEC	1	-80	10	0	1	Inf	0	15	4	0.8	GABA	4	0	0	10
MS-GABA to pBC	1	-80	1.333	0	1	Inf	0	2	4	0.8	GABA	1	0	0	10
MS-GABA to cBC	1	-80	2.667	0	1	Inf	0	2	4	0.8	GABA	1	0	0	10
MS-GABA to MC	1	-80	4	0	1	Inf	0	2	4	0.8	GABA	1	0	0	10
MS-GABA to HIPP	1	-80	4	0	1	Inf	0	2	4	0.8	GABA	1	0	0	10
MS-GABA to HICAP	1	-80	4	0	1	Inf	0	2	4	0.8	GABA	1	0	0	10
MS-GABA to ACh	1	-80	10	0	1	Inf	0	1	4	1	GABA	4	0	0	10

3. BIOLOGICAL SCALING OF NEUROGENESIS

Our primary effort in assessing our computational model of neurogenesis focused on examining the relationship of model scale on the overall impact of different rates of neurogenesis in response to familiar and novel information. While the presence of neurogenesis in the adult human hippocampus has been convincingly demonstrated and quantified using several methods ; the functional relevance of new neurons is heavily debated. Neurogenesis has been shown to have a clear relationship to both cognitive function and anti-depressant efficacy in rodent models . While these functions have been linked to the human dentate gyrus , the direct link of neurogenesis to these functions in humans has been limited to conjecture due to the impracticality of direct causal manipulation of the system. Further, because neurogenesis, like most neural processes, has mostly been characterized in animal models, the relatively low rates of neurogenesis coupled with the overall large scale of human networks has led to persistent skepticism of the cognitive and therapeutic implications of human hippocampal neurogenesis. To investigate the computational potential of neurogenesis at biologically realistic scales, we utilized both our high fidelity model and abstract models to ascertain how increase cell and synapse numbers affect computation.

3.1. Biologically Realistic Model Network Scaling

Most models of neurogenesis have used considerably reduced network sizes to study the effect of neurogenesis, and while these models have shown effects of neurogenesis, recently we posited that scaling models down would potentially occlude many of the potential effects of young neurons due to the effects of noise at small scales. To assess this, we performed a scaling study of DG networks for sizes from 1000 GCs (the network size in our previous study) to human sizes of 10,000,000 GCs (Figure 13). Our scaling was linear, which while not optimized is typical for most reduced biological network simulations. We further examined a range of neurogenesis rates, measured by the percentage of neurons at less than 2 months of age, from 0% (no neurogenesis) to the equivalent of about 1% a month .

First, in response to increasing network size, we observed that only in networks with substantial neurogenesis did we see maintenance of overall network activity. In contrast, in networks without neurogenesis we saw a substantially diminished firing rate. This pattern was clear in response to familiar inputs (Figure 13), but was particularly striking for novel inputs (Figure 13). This dropoff in activity was dramatic, with mature-only networks seeing a drop of activity by a factor of 10 in response to familiar information and nearly a factor of 100 in response to novel information.

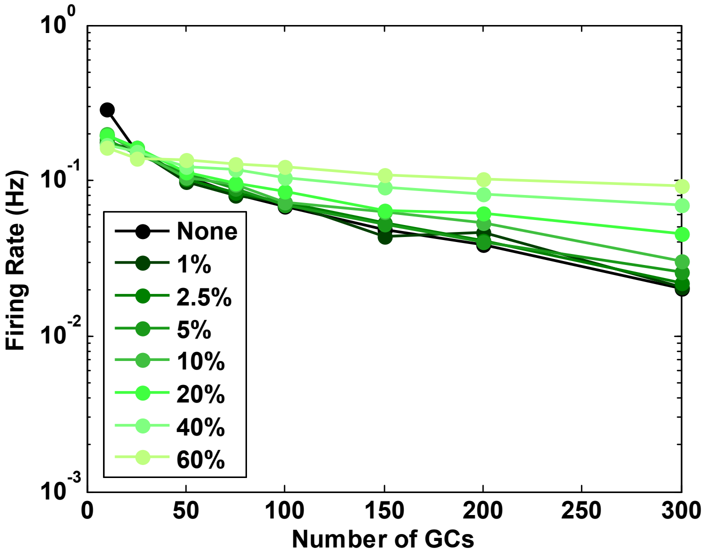


Figure 13 - Firing rate of GC layer in response to FAMILIAR inputs

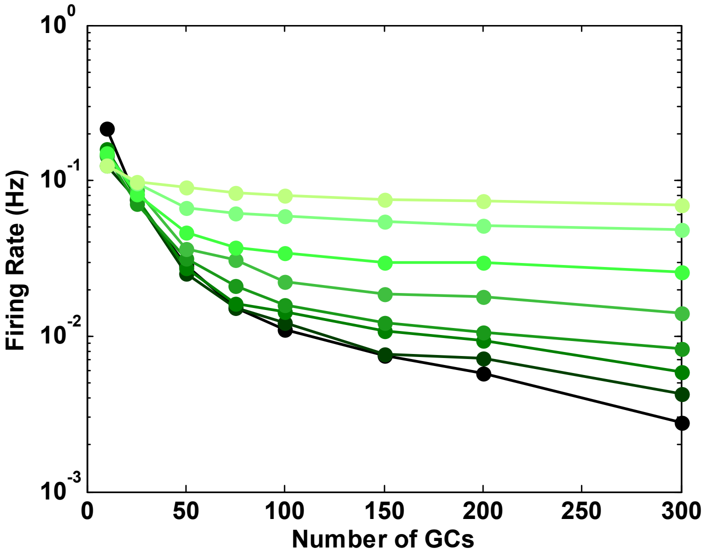


Figure 14 - Firing rate of GC layer in response to NOVEL inputs

Notably, we observed that this effect of neurogenesis on network activity is almost entirely due to the relatively high firing rates that immature neurons maintain regardless of overall network size. For instance, in a network of 25,000 neurons, in which all are mature, the overall firing activity can be seen to be fairly robust (Figure 14). If the network is neurogenic, and as such a subpopulation is immature, the overall activity is not all that different, and the relative contribution of young neurons is not immediately clear (Figure 15).

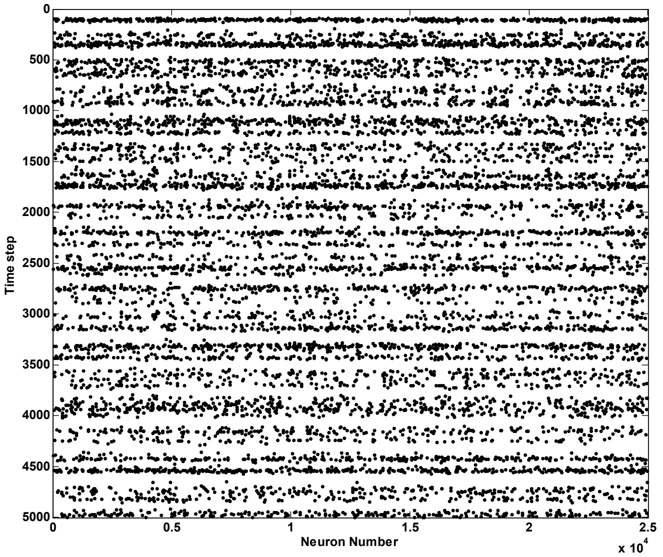


Figure 15 - Activity of GC layer (25,000 neurons) in no neurogenesis networks to familiar inputs

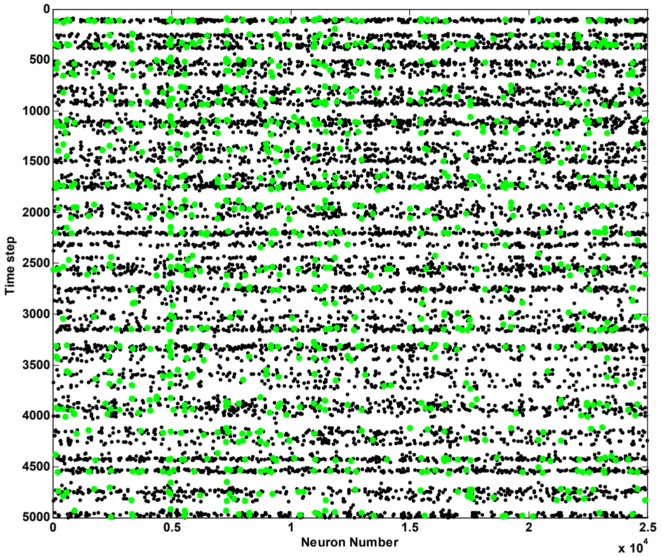


Figure 16 - Activity of GC layer (25,000 neurons) in neurogenic networks (green neurons = immature) to familiar inputs

On the other hand, if much larger networks are considered, wherein the overall activity is substantially lower due to the scale of the system, while the mature-only networks are quite silent (Figure 16), the effect of young neurons in neurogenic networks stands out considerably (Figure 17). Even though young neurons only make up a fraction of the overall population in each of these examples, they comprise a substantial fraction of the overall active population in realistic scale systems in response to both familiar (Figure 18) and novel inputs (Figure 19).

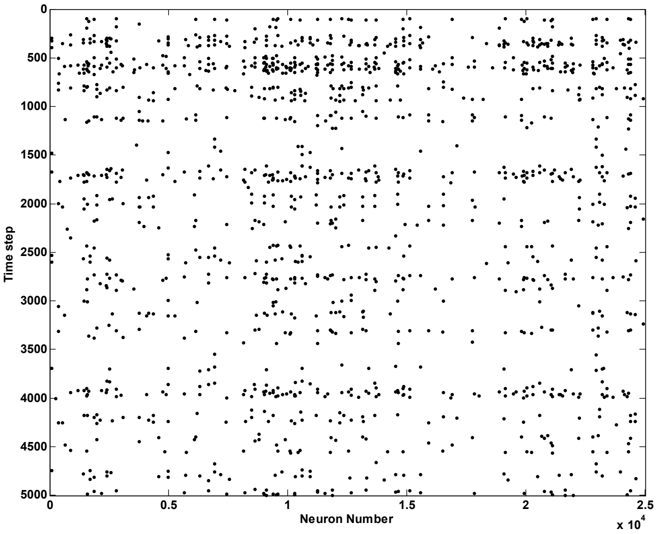


Figure 17 - Activity of GC layer (25,000 neurons out of 300,000 neurons) in no neurogenesis networks to familiar inputs

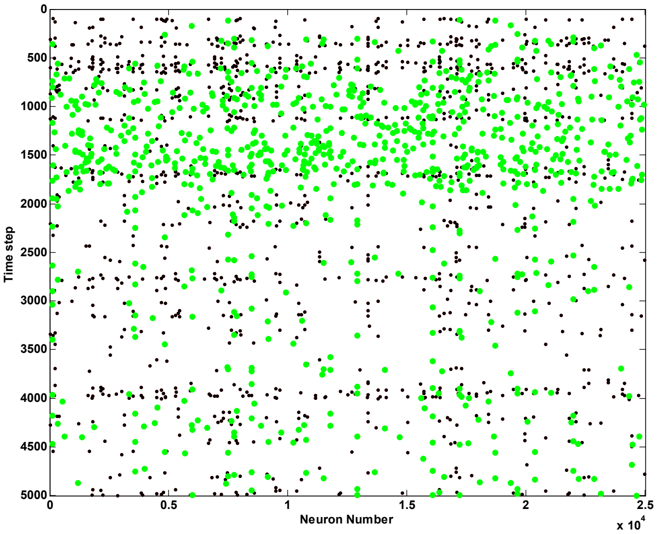


Figure 18 - Activity of GC layer (25,000 neurons out of 300,000 neurons) in neurogenic networks (green neurons = immature) to familiar inputs

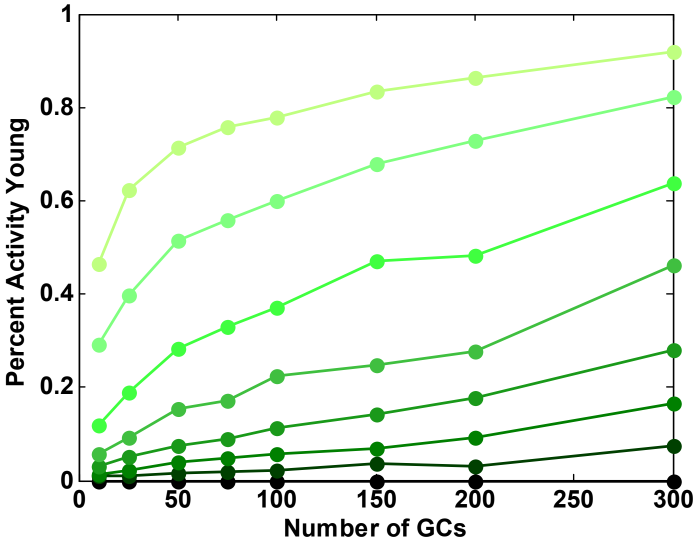


Figure 19 - Relative fraction of activity due to immature neurons in response to FAMILIAR inputs (see Figure 12 for legend)

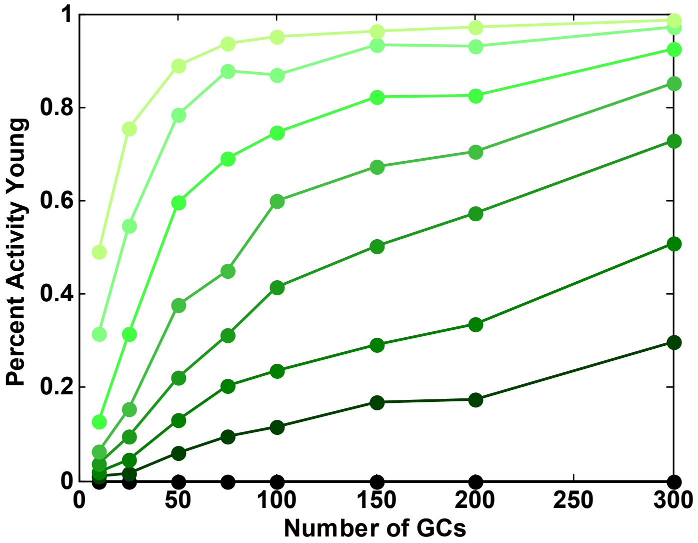


Figure 20 - Relative fraction of activity due to immature neurons in response to NOVEL inputs (see Figure 12 for legend)

3.2. Abstract model of network scaling

The observation that scale dramatically increases the potential impact of new neurons on DG computation is not necessarily intuitive. Indeed, across the full animal spectrum, as brains have evolutionarily gotten more sophisticated and, accordingly, larger, the overall extent of neurogenesis has decreased considerably. While neurogenesis in very simple invertebrates is not well characterized, it is quite pronounced in very simple vertebrate brains such as fish and reptiles. Birds have less neurogenesis than cold-blooded animals, however their neurogenesis appears somewhat more extensive than mammals, in which it is limited only a handful of regions including the DG. So why would neurogenesis be uniquely more critical in larger regions such as the DG in rodents? And further, why would it persist in humans and other animals whose brains are orders of magnitude larger?

One possible reason for the retention of neurogenesis in a dense region such as the DG is purely statistical. As the entry region of the hippocampus, which is responsible for episodic memory, the DG has a need to respond to both novel and familiar information. Getting a population of neurons to respond predictably and appropriately to novelty is actually a somewhat challenging task, as by definition novel inputs are previously unseen and thus cannot have been learned. This statistical challenge is fundamental to understanding why advanced cognitive neural systems operate at the scales in which they do; at small scales, as defined by the number of synapses on a typical neuron, noise in the distribution of synapses that are active at any given time is sufficient to drive neurons that are on average hyperpolarized due to inhibition. This noise driven activity is bad for the same reasons that are outlined in the introductory discussion about the balance of stability and plasticity in neural systems; if a neuron is trained to respond only to a subset of potential inputs, it should be robust against noise and not respond to non-trained events. However, the easiest solution for this – increasing the synaptic density such that noise is no longer sufficient to drive neuron activity – is actually detrimental to acquiring new information on those neurons onto which one wishes to train with novel events.

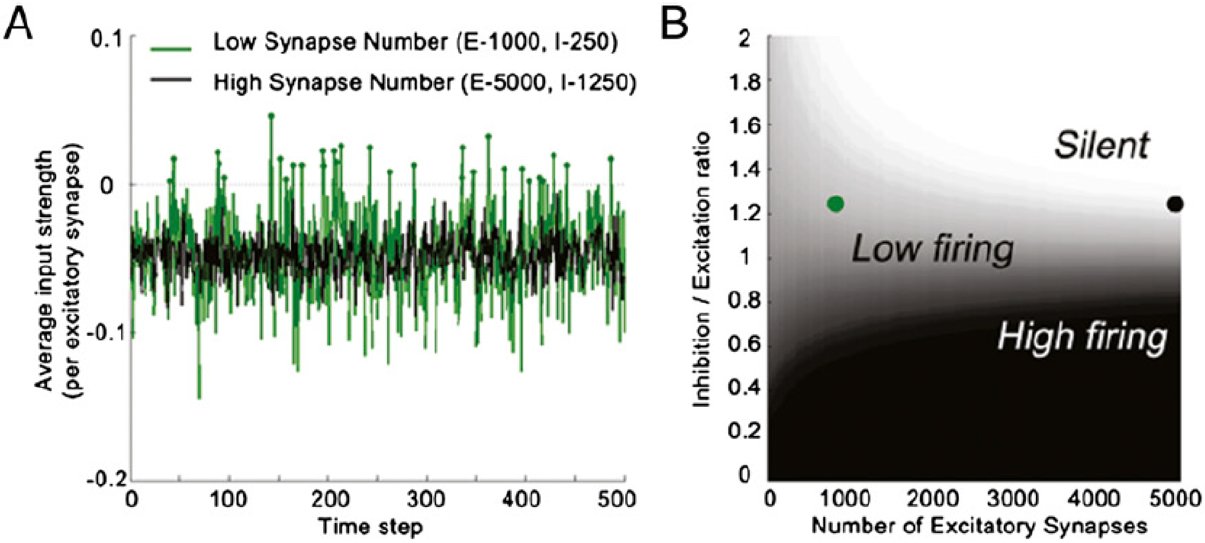


Figure 21 - Effect of synapse number on noise tolerance of neurons (from Li et al., 2012)

In an effort to continue exploring the effects of scaling on neural activity in a simple model framework, we worked with the Wadiche lab at University of Alabama-Birmingham to produce a simple model of the effects of neural synapse density on the dynamic range of pattern separation in a simple DG model. The model and results below are adapted from this collaborative effort.

3.2.1. Simple model methods (adapted from Deini et al, submitted)

<u>Estimating Perforant Path Connection Densities in Slice</u>

To estimate the number of intact synaptic connections onto mature and immature neurons in slice, we fit a basic statistical model to data provided by our collaborators describing the increasing correlation between a pair of neurons as excitatory drive current is provided. Given a pair of neurons, the number of expected shared inputs (N_{shared}; i.e., source fibers both neurons receive an input from) and independent inputs (N_{ind}; i.e., source fibers unique to one of the neurons) can be given by

$$N_{shared} = \frac{N_{total}}{N_{\infty}} \times N_{total} \tag{37}$$

$$N_{ind} = N_{total} - N_{shared} \tag{38}$$

where N_{total} is the total number of functional synapses on a GC (from that projection) and N_{∞} is the total number of potential input fibers. Supposing a stimulation of the fiber bundle activates a fraction p of the total inputs, one can derive the probability that both GCs are activated by the following equation based on binomial probabilities:

$$P_{overlap} = 1 - \left(1 - \left(1 - (1 - p)^{N_{ind}}\right)^{2}\right) \times (1 - p)^{N_{shared}} \tag{39}$$

where the left term represents the probability that both GCs are not independently receiving input from source fibers that are uniquely sampled and the right term represents the probability that no shared fibers are active.

Based on data from our collaborators, we used the constraint that the ratio of intact synapses on young neurons to mature neurons is 0.35. Further, we constrained the $N_{ind,mature}$ to be no more than five times $N_{shared,mature}$. Given these constraints and the measured values for mature and immature neurons in Figure 4, we performed a Monte Carlo search of 250,000 combinations of three independent parameters: $0.001<p<0.004$, $20<N_{ind,\ mature}<100$ and $100<N_{shared,\ mature}<500$, identifying which set of parameters produced a good fit with error defined as

$$err = \sqrt{\sum_{stim} \left(P_{estimated,stim} - P_{measured,stim}\right)^{2}} \tag{40}$$

where $P_{estimated,stim}$ is the output of equation (3) and $P_{measured,stim}$ refers to the measured overlap for a given stimulation level in Figure 4B. Notably, there were a number of solutions with approximately equivalent errors for which we selected p=0.0039; $N_{ind,mature}$=182; $N_{shared,mature}$=37, with a cumulative error (when compared to both immature and mature physiology data) of 0.1794.

Simple neural network model

We generated a simple perceptron-based neural network model of the entorhinal cortex (EC) to dentate gyrus (DG) circuit similar to that used in . The model consisted of 400 simple granule cell (GC) neurons and 1300 EC neurons (based on the estimate from our neuron fitting above). Each GC neuron was either considered mature or immature and randomly connected to neurons in the source EC population based on the frequencies determined above. Any connection resulted in a synapse of weight 1, and there was no learning or inhibition in the network.

For each trial, a fraction of EC neurons, EC_{act}, was randomly activated (set to equal 1). The downstream GC neurons were then considered active if their input surpassed their threshold, which was defined as a fraction of their synaptic inputs being co-active.

$$GC_{input} = EC \times W_{ECtoGC} \tag{41}$$

$$GC_{output} = \begin{cases} 1 & \text{if } GC_{input} \geq 0.2 \times N_{synapses} \\ 0 & \text{otherwise} \end{cases} \tag{42}$$

We tested each network on 100 sets of random EC inputs, and then computed the average overlap between GC outputs, which is given by

$$DG_{NDP} = \frac{1}{50 \times 99} \times \sum_{i=2}^{100} \sum_{j=1}^{i-1} \frac{GC_{output,i} \square GC_{output,\,j}}{\square GC_{output,i} \square \square GC_{output,\,j} \square} \tag{43}$$

We examined networks that were either 100% mature, 70% mature, or 100% immature. We further tested a series of different EC activation fractions.

3.2.2. Results of simple synaptic scaling model (adapted from Deini et al., submitted)

To isolate the contribution of excitatory drive, the model above did not include inhibition and we assumed that the same aged GCs have equivalent number of synapses that are sampled from the same total set of perforant path fibers. Furthermore, we did not consider probabilistic synapses. We constrained the ratio of excitatory drive (modeled as the number of synapses) to immature GCs according to the average ratio that we measured in simultaneous recordings of immature and mature GCs over a large range of stimulus intensities (0.35). Using a fitting approach described in the Methods, for MML stimulation, we observed strong fits with a $p_{minimal}$ of 0.39% and N_{∞} of 1,296 fibers. For mature GCs, we estimated an average of 219 MML inputs, of which 37 were shared between pairs of neurons, and for immature GCs we estimated 77 inputs, of which 5 were shared (note that this analysis is meant to replicate our experimental paradigm rather than to recapitulate synapse number based on authentic anatomical estimates). We then artificially generated networks with a hundred randomly generated neurons obeying these statistics, and observed that the randomly connected neurons exhibited comparable overlapped activation as observed experimentally (Figure 22).

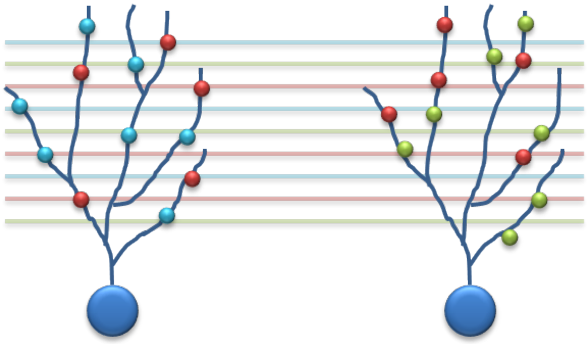


Figure 22 - Illustration of independent (blue and green) and shared (red) synapses

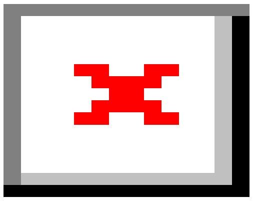


Figure 23 - Fit of synapse estimation model with real dual recording data

Next, we generated a simple network similar to that used in to emulate DG function (Figure 23), whereby different GC neurons had connection densities representative of either all mature, all immature or a mixture of 70% mature and 30% immature GCs. These networks shared input connection statistics comparable to the observed slice results; however each GC only fired if 20% of their inputs were active. Using this approach, we could observe how an increase in input similarity would affect the presumed pattern separation of random EC inputs.

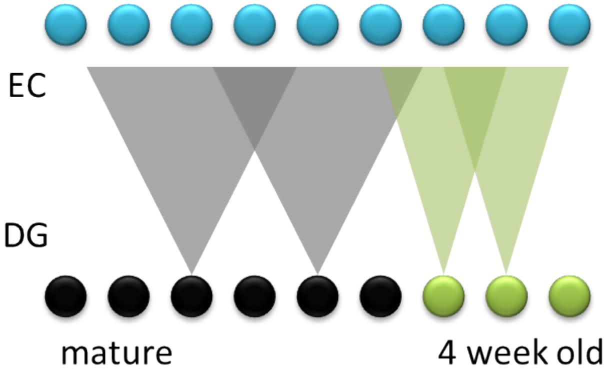


Figure 24 - Illustration of mixed immature and mature simple spiking model

Figure 25 - Increasing immature neuron population expands dynamic range

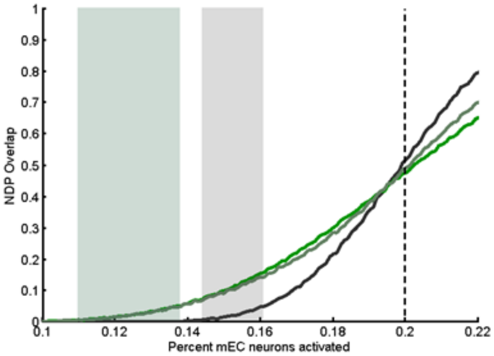


Figure 26 - Zoom in on expanded dynamic range

Consistent with previous modeling studies, networks with young neurons exhibited higher correlations (i.e. reduced pattern separation) than networks without neurogenesis so long as the input activity was on average below threshold for GCs to fire (Figure 24). However, the response curve of the mature only networks was quite steep; if EC activity was well below threshold, the mature-only DG was exceptional at pattern separation, but once threshold was approached, the network quickly became ineffective. In contrast, networks with immature neurons more gradually increased their activity as the overall input activity approached the neurons' threshold, suggesting that young neurons make the DG less sensitive to local changes in input statistics (Figure 24). In other words, addition of immature neurons with reduced excitatory drive was sufficient to expand the dynamic range of the dentate network for effective pattern separation.

Our results can be incorporated into a broader view of the DG's function in hippocampal coding . If the DG is to be relevant in driving CA3, it is necessary that its outputs have some low level of activity – perfect separation is meaningless if no information is communicated. Even a very low activity level necessitates some minimal level of neuronal overlap, however too much overlap presumably leads to interference in CA3 memory formation. In our simple model, maintaining GC activity within a range such that its correlations stay between 0.5% and 5% requires that the EC's activity level can only be tolerated within a range of about 1%. In contrast, the presence of immature neurons essentially doubles the permissible range of inputs, and further shifts the ideal range further away from the critical threshold level; minimizing the risk of incidentally stronger inputs of producing a severely disruptive signal (Figure 25). The presence of a subpopulation of neurons with fewer excitatory inputs is capable of increasing the dynamic range of the EC suitable to be processed by DG, suggesting network heterogeneity can provide a form of input normalization in addition to the effects of feed-forward and feedback inhibition in the circuit.

4. METRICS TO ASSESS NEUROGENESIS FUNCTION

The results described in the above section about the increased activity of young neurons is consistent with previous observations, but activity alone does not resolve the functional importance of neurogenesis. In particular, there is considerable debate about whether the activity of young neurons directly contributes to the DG pattern separation function or whether their impact is to increase information content in episodic memories (Figure 26). These two ideas have been discussed extensively in the literature, but there has been little attempt to develop metrics to quantify how an ensemble of neurons may provide either a pattern separation function or be increasing information content sent into the hippocampus. Most studies on the pattern separation ability of the DG have focused on looking at average correlations of the output population as compared to the correlations of an input layer. This reduction in similarity is typically measured without reference to the overall information encoded in the system. For example, if the EC contains substantial information about two events that may have occurred at a park, is it sufficient for the DG to simply encode only one feature that may have differed between the two events (say one had a dog involved and another a cat)? While this is potentially optimal from a pattern separation perspective, it would potentially trivialize the ultimate memory formed. Rather, the DG likely needs to maintain a balance between information coding and separation; effectively minimizing correlation between DG outputs is likely best thought of as a constraint as opposed to a function in and of itself.

This more sophisticated view of DG function requires both simultaneously assessing DG information content and correlations in a quantifiable, parametric sense. While there are a number of techniques to assess similarity (correlations, cosyne angles, Hamming distance, etc), the ideal metric is not immediately obvious, as this is likely going to be a function of how network similarity impact the downstream CA3 region. Further, the quantification of information content is a major challenge in systems neuroscience. Examining the nature of information representation in a model of thousands or millions of neurons requires the development of new approaches to ascertain the information content of large populations of neurons. While techniques exist for application of Shannon information approaches for single neurons, the primary challenge in quantifying information in a network is how to combine neurons, which are non-independent information channels .

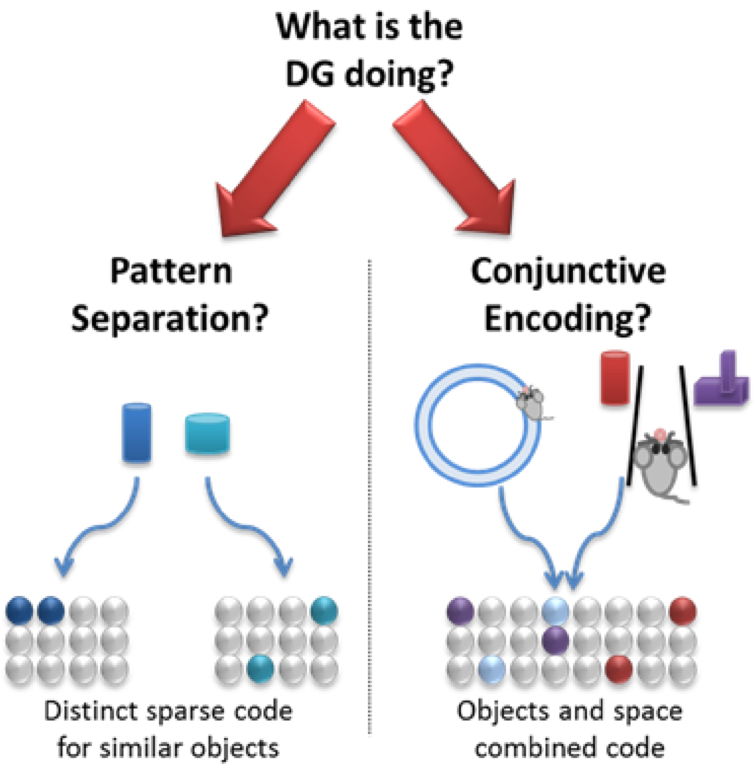

What is the DG doing?
Pattern Separation?
Conjunctive Encoding?
Distinct sparse code for similar objects
Objects and space combined code

Figure 27 - Illustration of two functions of DG

We approached this problem by identifying compression methods used broadly in computing as a potential route to quantify the redundancy in the network representation over time, thus providing an estimate for independent information communicated by a network. Compression can take a number of different forms. Here, we were concerned with *lossless* compression, which guarantees that the inputs are fully retrievable from the outputs (i.e., no information is lost in the compression process), at the possible expense of a suboptimal compression. The alternative - *lossy* compression – is widely used in file formats such as TIFF images and MP3 audio files works by allowing information that is not perceived by the end user to be eliminated as well, which typically results in a more compressed signal, but one which contains less information than the original source. For the purposes of our neural analysis, we are concerned with obtaining an estimate of total independent information content, which lossy compression will fail to provide.

There are a number of methods for lossless compression. Here, we explore two possibilities that can provide initial estimates for network redundancy and thus overall information content. The first is a linear method similar to principal components analysis (PCA). The second method is a digital compression approach known as Lempel-Ziv, which is widely used in file formats such as GIF image files and can be linked theoretically to Shannon entropy.

4.1. Linear compression analysis

Our first method used linear decomposition to estimate the extent to which our observed population activity was redundant. The idea behind linear compression is that identifying a reduced set of dimensions that represent the bulk of the data is much cheaper to transmit and store than the full data set. PCA works by rotating (and scaling) the dimensions of the original data set into a basis that maximally explains the variance of the data present originally. Those dimensions of this new basis set comprised of the principal components are then selected to represent the compressed signal.

The intuition behind using PCA on neural data can be described in a number of ways. Suppose a subpopulation of neurons is highly correlated in their activity, one way to more efficiently represent their activity is to imagine a single virtual neuron which responds as an average of that group. This virtual neuron can be thought of as the first principal component of the population. Next, a new second virtual neuron can then be generated that represents the difference between a subset of the original individual neurons' activities and the new virtual neuron's activity (the *residual* activity). Now these two virtual neurons, or two principal components, are orthogonal to each other and likely represent a good fraction of the overall network activity. This process can continue on until all the residual activity is accounted for in the virtual neurons. Because these virtual neurons are representing averages of many neurons' activities, they are effectively compressing much of the population's responses into a much smaller number of dimensions.

PCA can thus be used as a compression method; it is often common to ask how many of the top principal components are necessary to represent 95% of the original information. Generally, however, PCA is lossy – unless there are some entirely redundant initial channels, PCA will not allow the removal of dimensions without sacrificing some of the original information. However, by considering how much information was moved from the original dimensionality into the primary new dimensions - effectively how much compression PCA could provide, we contend that we can estimate how much redundancy is in the original data.

4.1.1. Method

Suppose **S** is an N x T matrix of the spiking output of a population of N neurons over T time steps. First, we collapse **S** into a temporally compressed N x t matrix **A**, where each column represents the binned firing output of the N neurons over sequential 25ms timesteps (t equals T/25), normalized to a mean of 0.

Let **V** equal a matrix of size t x t of all eigenvectors of the covariance matrix of **A**, which we call $\mathbf{A}^{\mathbf{T}}\mathbf{A}$, with their associated t eigenvalues $\{\lambda_i\}$ and let σ^2_i equal the variance of each column within **A** (variance of all neurons' firing over a given time bin). We will consider the cumulative variance of the network, which can be computed as

$$\sum_{i=1}^{t} \lambda_i = \sum_{i=1}^{t} \sigma_i^2 \qquad (44)$$

which simply indicates that the sum of the eigenvalues of $\mathbf{A}^{\mathrm{T}}\mathbf{A}$ is equivalent to the sum of diagonals of $\mathbf{A}^{\mathrm{T}}\mathbf{A}$, a property of symmetrical matrices.

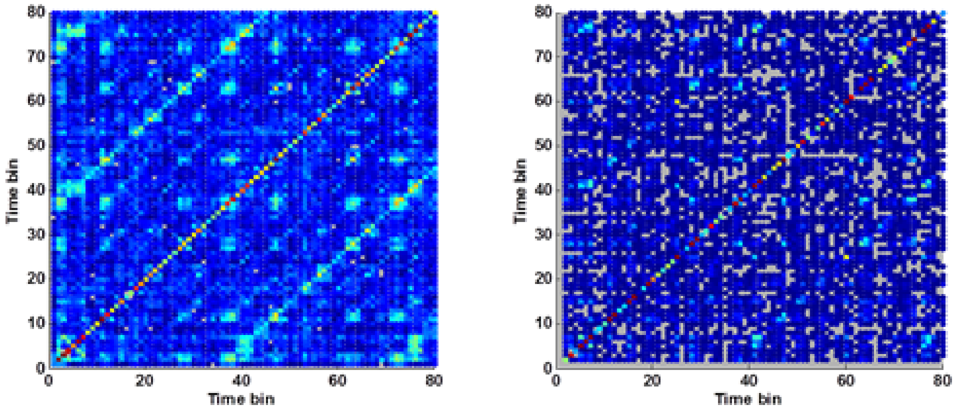


Figure 28 - Example covariance matrices of model activity. Left is EC, right is GC layer

This equivalence confers an interesting property with regard to measuring how redundant the component vectors of **A** are. Essentially, matrix decomposition transfers covariances between dimensions until that the new basis is orthogonal. The extent to which variance needs to be moved in order to obtain diagonality (the eigenvalue matrix) is an indicator of how orthogonal the original matrix was. By quantifying the difference between the distribution of λ's and σ^2's, we can estimate how much variance had to be transferred into other dimensions to orthogonalize the matrix. This is essentially a measure of how compressible (linearly) the original data is with PCA.

Let the vectors of **A** be ordered by increasing variance σ^2, and let the eigenvector matrices **D** and eigenvalues vector **λ** be ordered by increasing eigenvalues λ. Then, we can define a measure, Φ, as the estimated linear compressibility of the matrix as follows:

$$\Phi = 2\sum_{j=1}^{t}\left(\sum_{i=1}^{j}\sigma_i^2 - \sum_{i=1}^{j}\lambda_i\right)\Big/\sum_{i=1}^{t}\sigma_i^2 t \qquad (45)$$

The extremes are illustrative. If **A** is orthogonal, $\mathbf{A}^T\mathbf{A}$ will be diagonal and the eigenvalues λ's will be equivalent to the variances, σ^2's, of **A**. In this case, Φ will be zero, indicating that A is not compressible and fully separate. At the other extreme, if **A** is entirely redundant such that the signal does not change from one time bin to the next, then Φ will approach one (specifically, 2*(t-1)/t) as the number of independent time vectors are increases. If Φ can be taken as a measure of the linear redundancy of **A**, then the extent of the total activity of **A**, f_A, that is not redundant is a metric of independent information represented by **A**, which we define as Ψ.

$$\psi = (1 - \square) \times f_A \qquad (46)$$

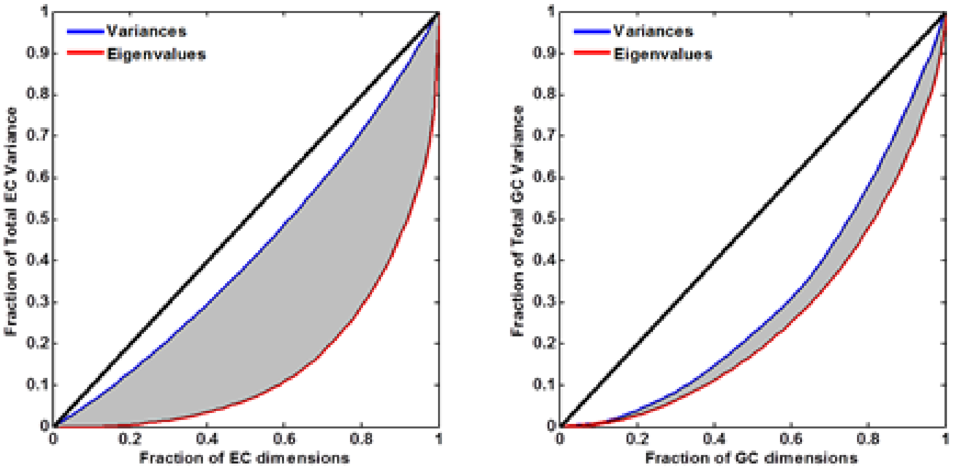


Figure 29 - Example principal components compression estimation. Left is EC, right is GC

4.1.2. Results

We analyzed both pattern separation and memory resolution in the neurogenic and non-neurogenic simulations at different scales by quantifying the proportion of redundant variance in the GC outputs in response to familiar and novel inputs (**Figure 2f-g**). Notably, increased levels of neurogenesis did not appreciably increase correlations in GC activity (i.e., did not impair pattern separation) by this measure; an observation that differs from our previous model which suggested that young neurons increased coding similarity, a process we referred to as *pattern integration* . In contrast, it appears that neurogenesis slightly improved overall network orthogonalization, a result somewhat consistent with the pattern separation interpretation of behavioral results. Further, the addition of neurogenesis greatly increased the non-redundant communication by the DG, a feature which would be expected to subsequently improve the ability for behavioral discrimination of memories **(Figure 2h-i)**. These two results suggest that the behavioral improvements through neurogenesis are due to its facilitation of increased memory resolution while preserving the DG's overall pattern separation role.

4.2. Digital Compression

4.2.1. Lempel-Ziv & Normalized Complexity Analysis

Lempel-Ziv (LZ) coding is an online, adaptive class of techniques for source coding . It consists of adaptive dictionary compression algorithms which are universally optimal in that their asymptotic compression rate approaches the entropy rate of the source for any stationary ergodic source . Rather than building an optimal coding based upon known a priori knowledge of the frequency of occurrence of the symbols being encoding (such as Huffman coding does), the LZ algorithm parses a string and builds dictionary entries based upon the shortest phrase not yet seen. Repeated sub-strings result in larger dictionary entries, so effectively the LZ algorithm is able to dynamically generate more efficient representations for the most prevalent sub-strings.

Applied to the neural domain, this approach allows us to analyze the encoding of neural ensembles without knowing firing behavior probability distributions of each neuron. Rather, we have used complexity as a measure of compressibility in order to estimate entropy to quantitatively assess the information content of a signal. Szczepanski et al. applied the general Lempel-Ziv complexity (LZ-Complexity) measure to estimate entropy of real and simulated neurons . But unlike the work of Szczepanski et al., rather than applying LZ-Complexity analysis to an individual neuron's spike train; we have applied the approach to a neural population as a whole. LZ-Complexity is based upon measuring the rate of generation of new patterns along a sequence of characters in a string being compressed . Applied to neuron spike trains, this technique looks for repeated spiking behavior over time. Instead, by applying it across an entire neural ensemble, we assessed repeated patterns of neural activity.

We have explored a couple of approaches to analyzing the multidimensional signal comprised of an ensemble of neurons firing over time. Our first approach was to take the co-activity of all neurons in the ensemble at an instance in time and concatenate each of these temporal segments into a single spike signal. This approach is depicted in Figure 29.

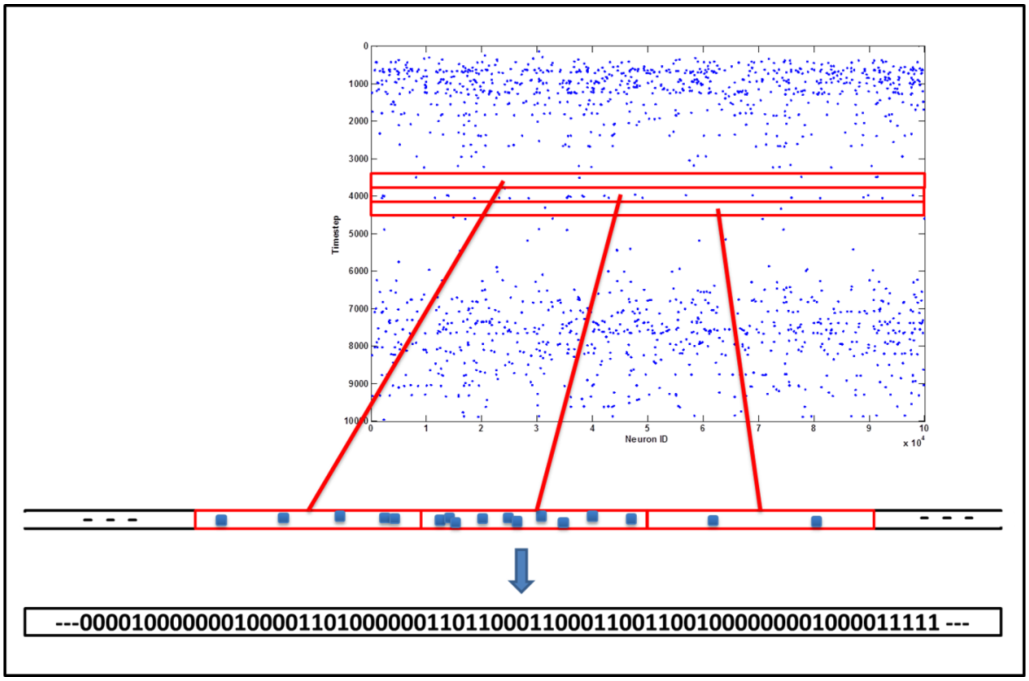


Figure 30 - Concatenation of neural firings across the population ensemble to generate a binary spike signal preserving temporal synchrony

Alternatively, rather than concatenating each segment into a single spike signal we also investigated a piecewise analysis in which a single segment (whether temporal or ensemble) at a time is passed to the dynamically expanding dictionary. An illustration of this approach is shown in Figure 30. Using this piecewise approach, the same dictionary is repeatedly utilized and updated with each segment presented. However, dictionary entries cannot span different segments (as is the case with the concatenated single spike signal).

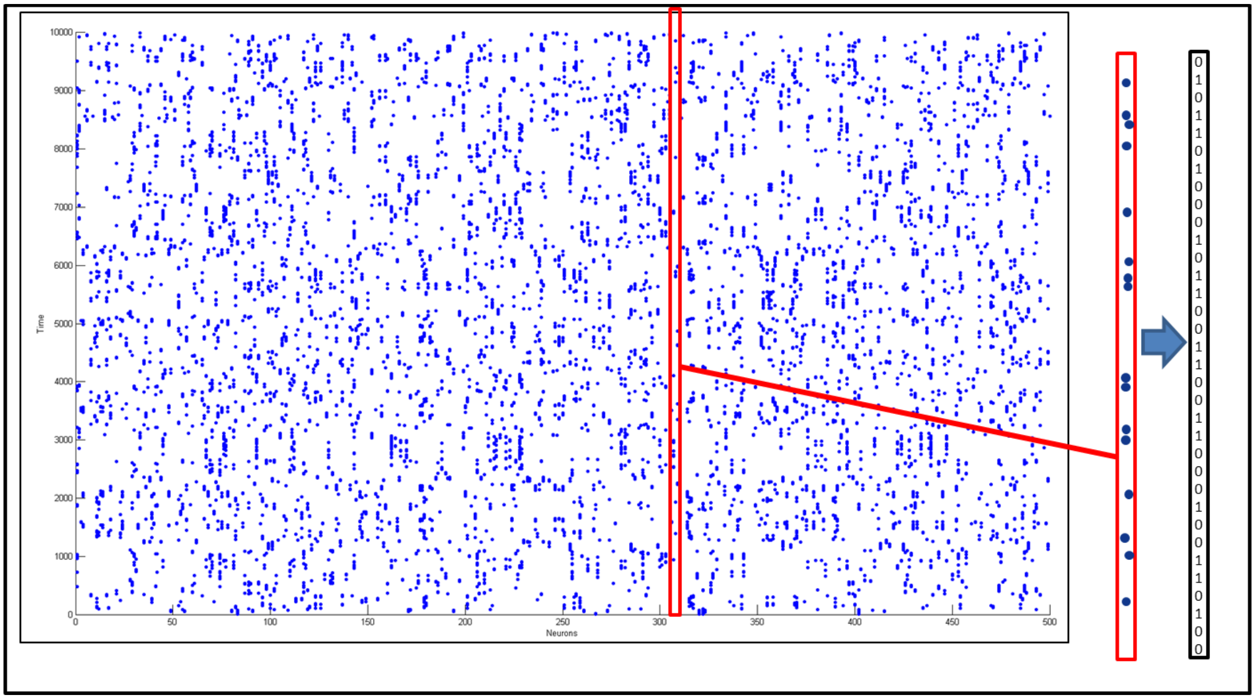


Figure 31 - Temporal piecewise segmentation of of neural firings

Regardless of which approach is used, the spike signal is converted into a binary signal where an action potential is encoded as a one and the absence of activity by a zero. The normalized complexity may then be computed as follows:

$$c_{\alpha}(x^{n}) = \frac{C_{\alpha}(x^{n})}{n} * \log_{\alpha}(n) \tag{47}$$

Normalized complexity measures the generation rate of new patterns along a word of length *n* with letters from an alphabet of size α (in this case two). Additionally, it can be proven that as the string length (our series of neural firings in this case) goes to infinity, the supremum of the normalized complexity approaches the entropy of the signal S:

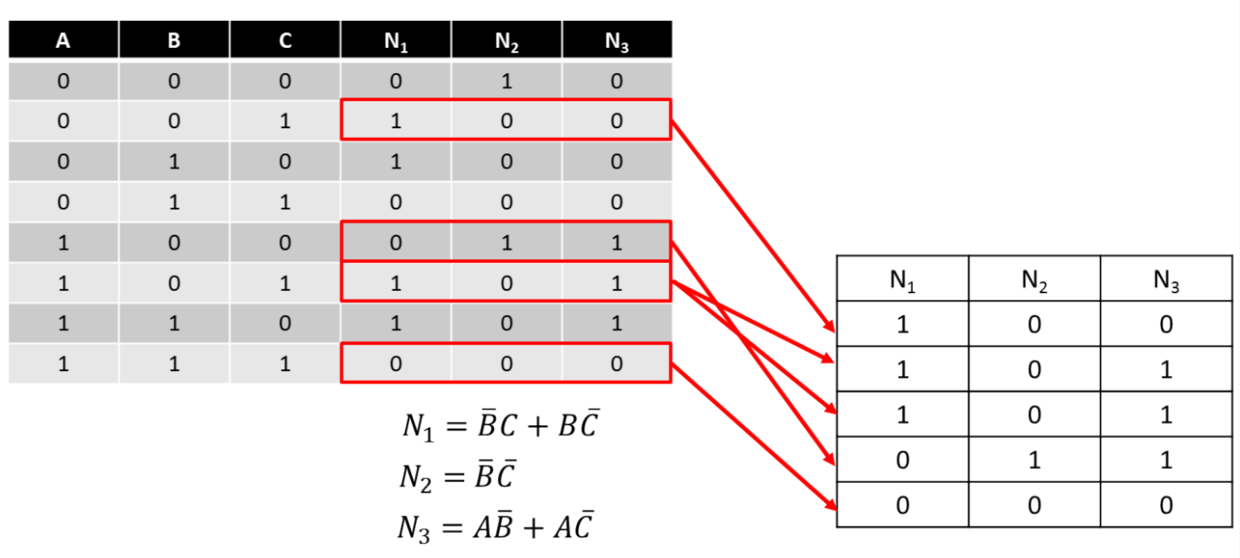
$$\lim_{n\to\infty} \sup c_{\alpha}(x^{n}) \leq H_{\square}(S)\square \tag{48}$$

Consequently, this provides us with a technique to approximate the information content encoded within a neural ensemble as expressed by the firing behavior the neurons exhibit.

4.2.2. Approximate Function Understanding Through Sampling

As a simplification overlooking the vast intricacies involved in their operation, neural behavior may be described as a function. Neurons fire in response to input stimuli which are able to drive the potential of the neuron beyond threshold. Ignoring the complexities of various learning mechanisms which facilitate the dynamic modification of neural responses to a given stimuli, the behavior of a neuron is a functional response. Each neuron yields a mapping from inputs to its functional output (namely whether or not to fire). As a simple, yet related, mathematical expression consider the canonical Boolean function. As the fundamental basis of digital logic and computing Boolean functions describe the behavior the function exhibits over all possible binary inputs. This is typically represented by a truth table such as that shown in the upper left of Figure 31. In this example, there are three inputs (A, B, and C) which allows for eight possible binary permutations. Three arbitrary Boolean functions of the inputs are defined by the columns N_1, N_2, and N_3. In a small idealized scenario such as this, it is tractable to furthermore specify the minimal functional representation as well.

In the neural domain this idealized analysis is not possible. In general neurons typically have 10,000 input connections. Even for a single neuron with binary synaptic connectivity, it is not tractable to consider $2^{10,000}$ possible unique input permutations. Rather, whether in a computational neural model or physiology recordings one can only sample the neural response behavior over a small subset of the possible space. This is analogous to only sampling a subset of the full Boolean truth table such as the selection of the red boxes leading to the breakout table show in Figure 31.



A	B	C	N_1	N_2	N_3
0	0	0	0	1	0
0	0	1	1	0	0
0	1	0	1	0	0
0	1	1	0	0	0
1	0	0	0	1	1
1	0	1	1	0	1
1	1	0	1	0	1
1	1	1	0	0	0

$$N_1 = \bar{B}C + B\bar{C}$$
$$N_2 = \bar{B}\bar{C}$$
$$N_3 = A\bar{B} + A\bar{C}$$

N_1	N_2	N_3
1	0	0
1	0	1
1	0	1
0	1	1
0	0	0

Figure 32 - Truth table for three input Boolean functions and an observational sampling

Without complete knowledge there are limits to what information may be inferred from this sampling. For example, in conjunction the three Boolean functions (N_1, N_2, N_3) have five unique response patterns over all possible inputs (000, 010, 011, 100, and 101). But the arbitrary sampling shown only captures four of the five possibilities and cannot completely infer the full functionality encoded by these Boolean functions. Rather, one can only estimate the behavioral properties of the combined function. Compression analysis, as previously described, is one means of inferring the complexity of the functionality and in effect estimating the encoding of the composite function. How well the functionality may be inferred also depends upon the sampling provided.

This is the exact same limitations imposed by the neural domain. Rather than having absolute knowledge of the inputs and outputs of all neurons over the full set of possible permutations, instead a typical neural recording is analogous to the breakout table in Figure 31.

4.2.3. Control Study Experimental Paradigm

As a control study to investigate the accuracy of our information estimation method, we have implemented an experimental paradigm which allows us to vary neural input resolution to control what the information content of the ensemble should be. Our analysis technique is a general technique and not specific to any neural region, but for our experimental paradigm we are examining information encoding within the dentate gyrus (DG), with the input to the DG comes from entorhinal cortex (EC. Grid cells of the EC encode a path trajectory through space and serve as the inputs to the DG place cells. By varying the precision of the DG grid cell encoding of the input space we can define to what fidelity the neural encoding is able to distinguish the input path. This notion is captured in the upper portion of Figure 32. The top square of the figure illustrates an arbitrary trajectory through space. The middle three squares portray DG grid cell encodings of various fidelities. As can be seen in the leftmost square with a resolution of four (partitioning the space into fourths along each dimension), this low resolution cannot distinguish between the twists and turns of the trajectory. Rather, anytime the input path lies within the region that grid is activated. Conversely, the higher resolution partitioning of the space (such as that shown by the other two samples) encodes a higher fidelity representation of the original path allowing it to distinguish precision such as the large loop in the upper right quadrant. As a result, the increased resolution provides more information in the sense that you are more certain about the precise position in physical space.

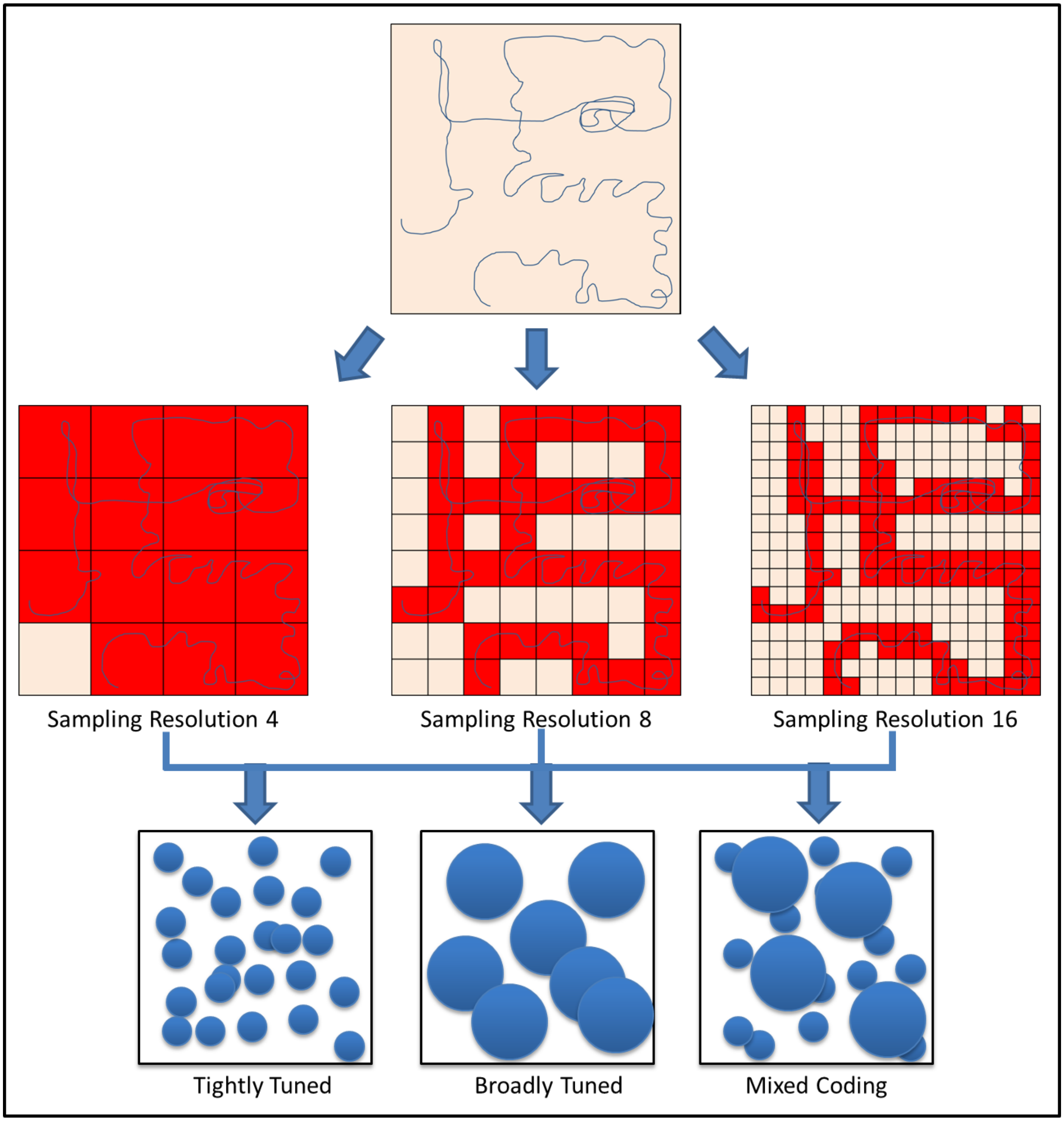


Figure 33 - Control Study Experimental Paradigm

The various grid cell encodings then serve as the input to DG place cells whose firing activity are the neural samplings we are interested in analyzing for information content. In this paradigm, the DG place cells are randomly placed topologically. And additionally, each place cell has a field width parameter which dictates the extent of which inputs are able to drive it to fire. The bottom portion of Figure 33 illustrates various configurations of place cell widths corresponding to encoding schemes. The leftmost example corresponds to a neural ensemble consisting entirely of mature, tightly tuned neurons that respond to precise inputs. Conversely, the middle example illustrates a neural ensemble consisting solely of broadly tuned neurons which respond to a broad set of inputs. The rightmost example portrays a mixed coding scheme, such as that hypothesized by neurogenesis where young neurons are broadly tuned and hyper receptive, but as they mature become tightly tuned to respond to specific inputs.

This experimental paradigm allows us to generate neural spiking outputs which we can then run our compression techniques on to approximate the information content in a controlled manner.

4.2.4. Results

For our assessments we have primarily analyzed an ensemble of 500 randomly placed neurons over 10,000 timesteps. Figure 33depicts one path intended to loosely resemble Brownian motion across the space. Fig. 6 illustrates the resulting firing rates of these neurons over this path for resolutions of 4 and 100. With fairly narrow place cell widths, only neurons closest to the active grid location are driven to fire. Consequently, as can be seen in the top half of the figure, the neurons which happen to be located central to grid positions with the coarse binning resolution of four fire frequently. Alternatively when the resolution is much finer, such as captured in the lower half of the figure, a lot more neurons fire, but less frequently. In this sense, the neural encoding of the path is distributed across multiple neurons and has the ability to represent more information.

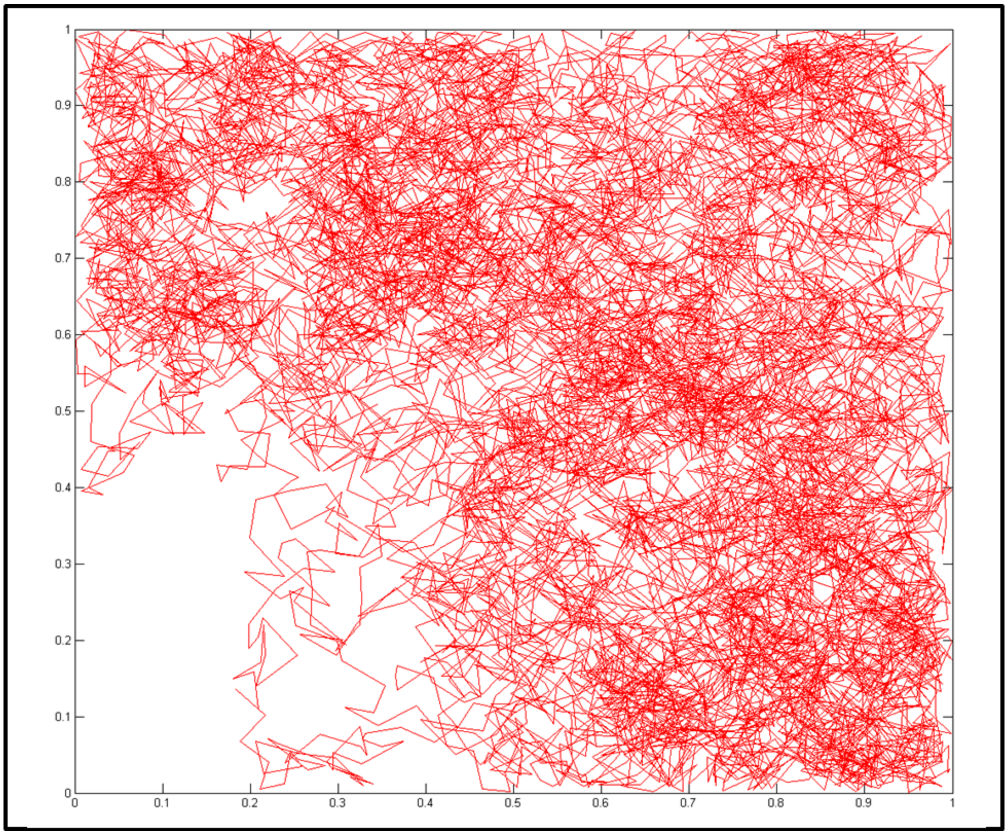


Figure 34 - Sample random path over 10,000 timesteps

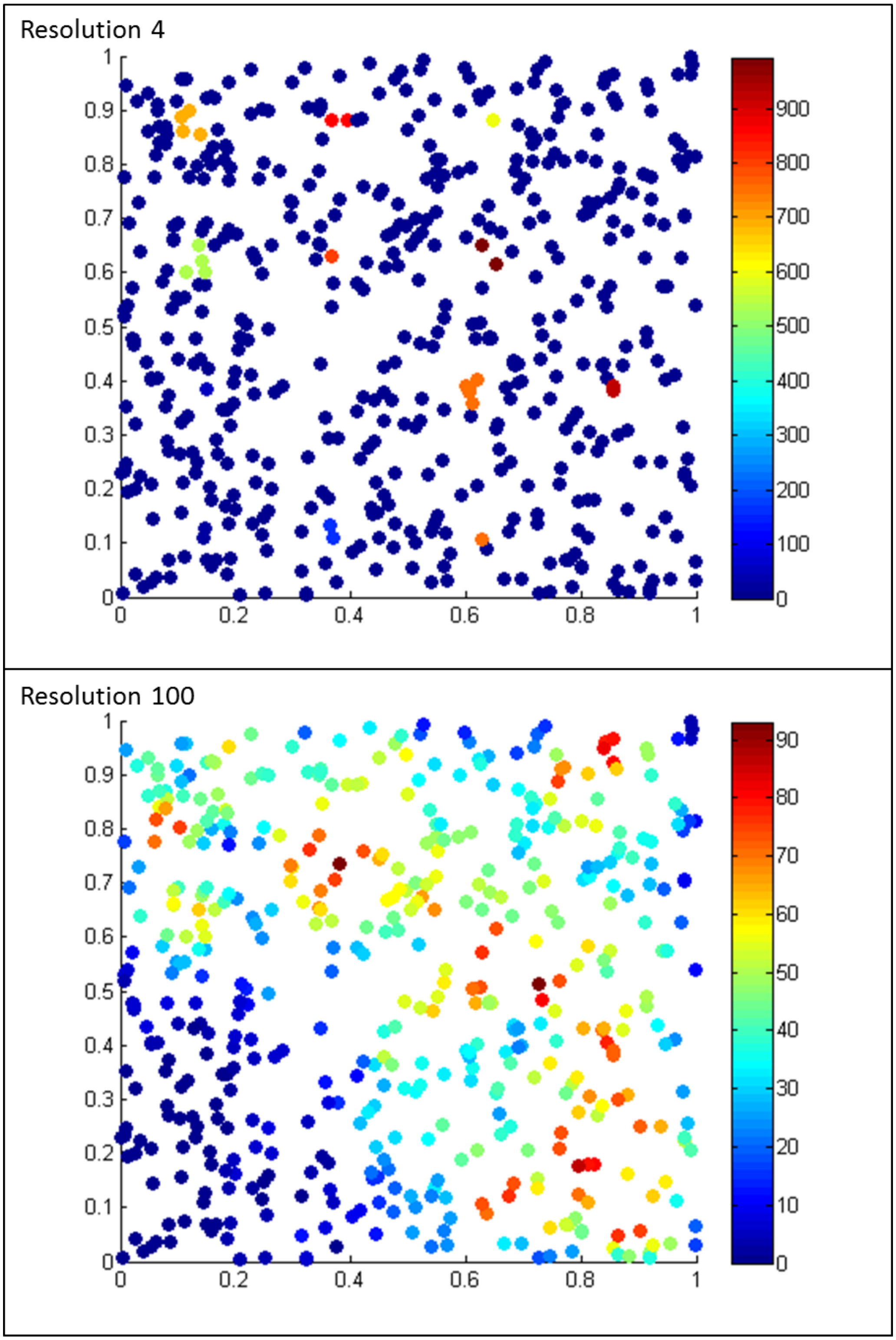


Figure 35 - Ensemble firing rates for resolutions 4 and 100

Keeping the neuron positions, path, and place cell widths fixed for a given analysis we have then varied the resolution and estimated the information content based upon the observed neural firings. The resolution of the EC grid cell encoding provides a theoretical upper bound as to what precision the compression analysis approximation can estimate - namely log_2(resolution^2). However, not all paths fully cover the space nor do they cover each region uniformly. And so a more precise theoretical upper bound may be computed for a specific path by using the frequency of occurrence of a particular resolution region as the probability of the event that that particular region will be active. Treating the regions of space as outcomes, these frequentist inferences allow us to calculate the actual entropy of a specific path through space as a straightforward calculation.

Applying this compression analysis of neural information content yields the approximations captured in Figure 35compared with the theoretical information content upper bound (shown in red). Across the x-axis are increasing resolutions from 1 to 10, and the y-axis is the information estimate (entropy).

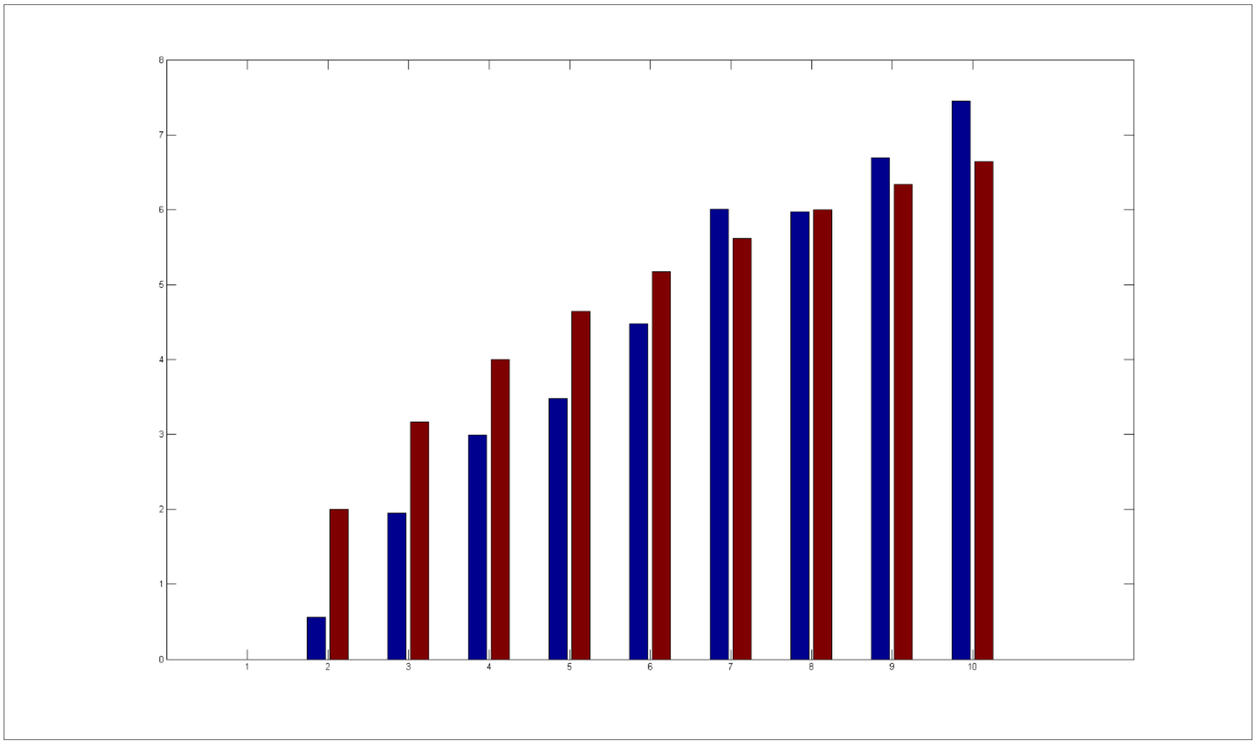
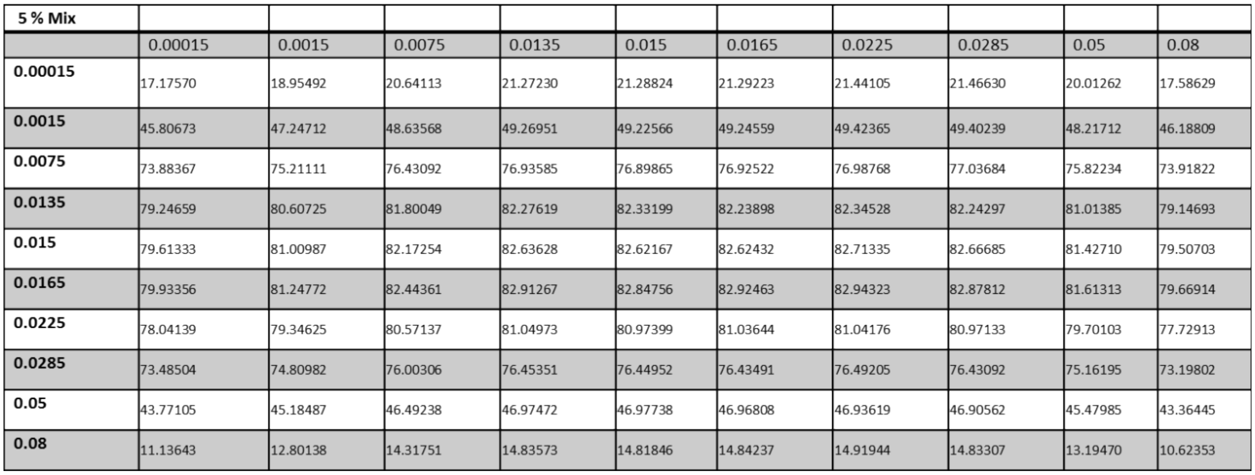


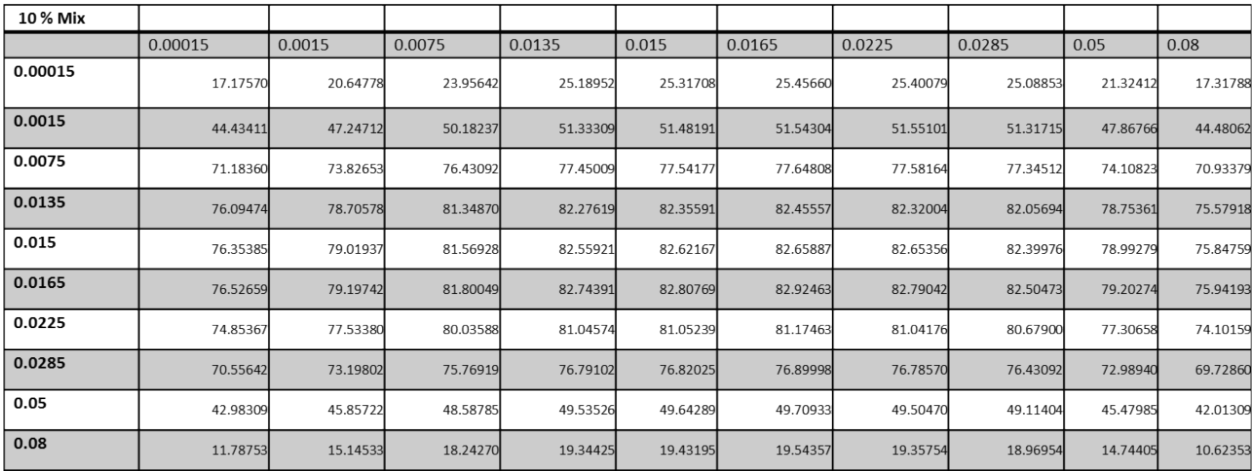
Figure 36 - Ensemble Entropy vs. Theoretical Upper Bound

While the normalization technique needs to be modified to account for our piecewise LZ presentation approach, the general trend exhibited by our estimations appears promising. In the absence of a corrected normalization technique the values themselves are not meaningful, but we have also applied the approach to ensembles of mixed place cell widths to investigate the effect on information content with a mixed coding scheme. As follows are results of varying the mix ratio all with a path resolution of 10 and fixed neuron positions and path for all experiments. In each of the charts, the place cell width listed along the rows is the majority cell type with the resolution at the column intersection corresponding to the mixed in minority cell type. For example, in a given row column intersection with a five percent mix there will be 475 neurons with place cell widths given by the row and 25 neurons with that given by the column. The main diagonal of each table corresponds to a pure coding at a given resolution.



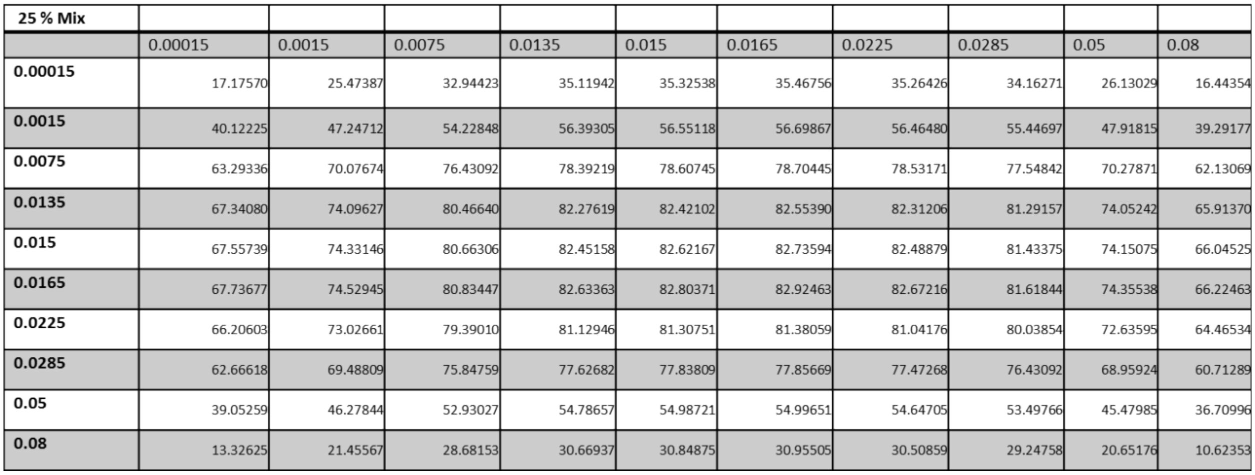
5 % Mix										
	0.00015	0.0015	0.0075	0.0135	0.015	0.0165	0.0225	0.0285	0.05	0.08
0.00015	17.17570	18.95492	20.64113	21.27230	21.28824	21.29223	21.44105	21.46630	20.01262	17.58629
0.0015	45.80673	47.24712	48.63568	49.26951	49.22566	49.24559	49.42365	49.40239	48.21712	46.18809
0.0075	73.88367	75.21111	76.43092	76.93585	76.89865	76.92522	76.98768	77.03684	75.82234	73.91822
0.0135	79.24659	80.60725	81.80049	82.27619	82.33199	82.23898	82.34528	82.24297	81.01385	79.14693
0.015	79.61333	81.00987	82.17254	82.63628	82.62167	82.62432	82.71335	82.66685	81.42710	79.50703
0.0165	79.93356	81.24772	82.44361	82.91267	82.84756	82.92463	82.94323	82.87812	81.61313	79.66914
0.0225	78.04139	79.34625	80.57137	81.04973	80.97399	81.03644	81.04176	80.97133	79.70103	77.72913
0.0285	73.48504	74.80982	76.00306	76.45351	76.44952	76.43491	76.49205	76.43092	75.16195	73.19802
0.05	43.77105	45.18487	46.49238	46.97472	46.97738	46.96808	46.93619	46.90562	45.47985	43.36445
0.08	11.13643	12.80138	14.31751	14.83573	14.81846	14.84237	14.91944	14.83307	13.19470	10.62353

Figure 37 - 5% Mixed Coding Results



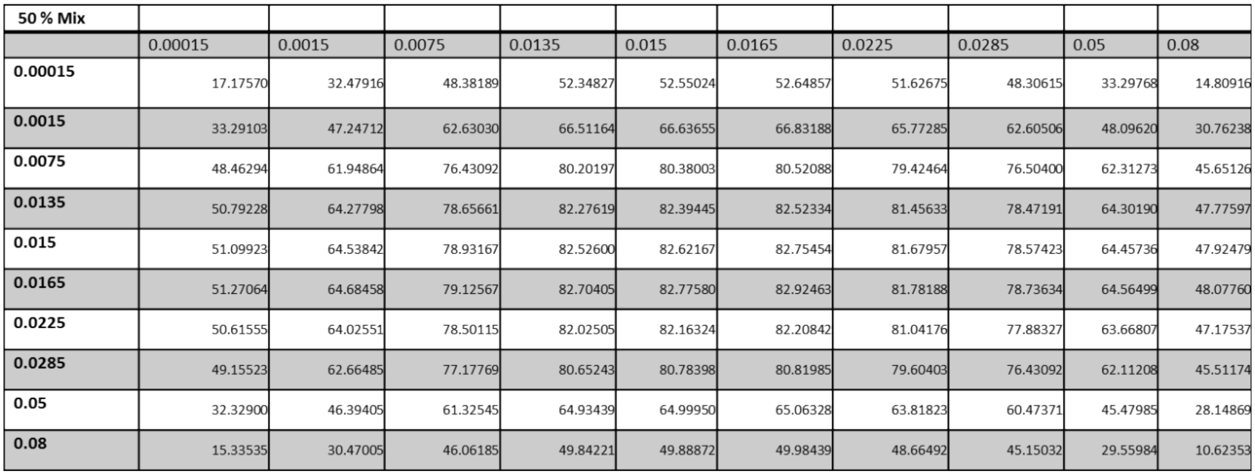
10 % Mix										
	0.00015	0.0015	0.0075	0.0135	0.015	0.0165	0.0225	0.0285	0.05	0.08
0.00015	17.17570	20.64778	23.95642	25.18952	25.31708	25.45660	25.40079	25.08853	21.32412	17.31788
0.0015	44.43411	47.24712	50.18237	51.33309	51.48191	51.54304	51.55101	51.31715	47.86766	44.48062
0.0075	71.18360	73.82653	76.43092	77.45009	77.54177	77.64808	77.58164	77.34512	74.10823	70.93379
0.0135	76.09474	78.70578	81.34870	82.27619	82.35591	82.45557	82.32004	82.05694	78.75361	75.57918
0.015	76.35385	79.01937	81.56928	82.55921	82.62167	82.65887	82.65356	82.39976	78.99279	75.84759
0.0165	76.52659	79.19742	81.80049	82.74391	82.80769	82.92463	82.79042	82.50473	79.20274	75.94193
0.0225	74.85367	77.53380	80.03588	81.04574	81.05239	81.17463	81.04176	80.67900	77.30658	74.10159
0.0285	70.55642	73.19802	75.76919	76.79102	76.82025	76.89998	76.78570	76.43092	72.98940	69.72860
0.05	42.98309	45.85722	48.58785	49.53526	49.64289	49.70933	49.50470	49.11404	45.47985	42.01309
0.08	11.78753	15.14533	18.24270	19.34425	19.43195	19.54357	19.35754	18.96954	14.74405	10.62353

Figure 38 - 10% Mixed Coding Results



25 % Mix										
	0.00015	0.0015	0.0075	0.0135	0.015	0.0165	0.0225	0.0285	0.05	0.08
0.00015	17.17570	25.47387	32.94423	35.11942	35.32538	35.46756	35.26426	34.16271	26.13029	16.44354
0.0015	40.12225	47.24712	54.22848	56.39305	56.55118	56.69867	56.46480	55.44697	47.91815	39.29177
0.0075	63.29336	70.07674	76.43092	78.39219	78.60745	78.70445	78.53171	77.54842	70.27871	62.13069
0.0135	67.34080	74.09627	80.46640	82.27619	82.42102	82.55390	82.31206	81.29157	74.05242	65.91370
0.015	67.55739	74.33146	80.66306	82.45158	82.62167	82.73594	82.48879	81.43375	74.15075	66.04525
0.0165	67.73677	74.52945	80.83447	82.63363	82.80371	82.92463	82.67216	81.61844	74.35538	66.22463
0.0225	66.20603	73.02661	79.39010	81.12946	81.30751	81.38059	81.04176	80.03854	72.63595	64.46534
0.0285	62.66618	69.48809	75.84759	77.62682	77.83809	77.85669	77.47268	76.43092	68.95924	60.71289
0.05	39.05259	46.27844	52.93027	54.78657	54.98721	54.99651	54.64705	53.49766	45.47985	36.70996
0.08	13.32625	21.45567	28.68153	30.66937	30.84875	30.95505	30.50859	29.24758	20.65176	10.62353

Figure 39 - 25% Mixed Coding Results



50 % Mix										
	0.00015	0.0015	0.0075	0.0135	0.015	0.0165	0.0225	0.0285	0.05	0.08
0.00015	17.17570	32.47916	48.38189	52.34827	52.55024	52.64857	51.62675	48.30615	33.29768	14.80916
0.0015	33.29103	47.24712	62.63030	66.51164	66.63655	66.83188	65.77285	62.60506	48.09620	30.76238
0.0075	48.46294	61.94864	76.43092	80.20197	80.38003	80.52088	79.42464	76.50400	62.31273	45.65126
0.0135	50.79228	64.27798	78.65661	82.27619	82.39445	82.52334	81.45633	78.47191	64.30190	47.77597
0.015	51.09923	64.53842	78.93167	82.52600	82.62167	82.75454	81.67957	78.57423	64.45736	47.92479
0.0165	51.27064	64.68458	79.12567	82.70405	82.77580	82.92463	81.78188	78.73634	64.56499	48.07760
0.0225	50.61555	64.02551	78.50115	82.02505	82.16324	82.20842	81.04176	77.88327	63.66807	47.17537
0.0285	49.15523	62.66485	77.17769	80.65243	80.78398	80.81985	79.60403	76.43092	62.11208	45.51174
0.05	32.32900	46.39405	61.32545	64.93439	64.99950	65.06328	63.81823	60.47371	45.47985	28.14869
0.08	15.33535	30.47005	46.06185	49.84221	49.88872	49.98439	48.66492	45.15032	29.55984	10.62353

Figure 40 - 50% Mixed Coding Results

In addition to applying this compression analysis approach to our control study experimental paradigm, we have also applied the concatenated string of ensemble firings approach to large scale neural models. See for more information.

4.2.5. Conclusions & Next Steps

We have presented a general approach to analyze the encoding of neural ensembles by using complexity as a measure of compressibility in order to estimate entropy to quantitatively assess the information content of a signal. Variations to this class of compression analysis of neural information content include:

- Contiguous string vs. piecewise segments
- Ensemble & Temporal segments
- Pre-populating the LZ dictionary with all possible zero permutations to focus only upon firing
- Mixed coding ensembles

Our next steps for this research include:

The LZ77 algorithm (Sliding Window) may yield different results than LZ78. In characterizing the application of the LZ class of algorithms will there be benefits to applying each in different scenarios such as ensemble vs. temporal segments? The discrepancies between these algorithms needs to be investigated as well as the potential application of other online compression algorithms.

- Performing sensitivity analysis sweeps over parameters such as the random path, neuron positions, and place cell resolutions.
- Investigate whether there is any fundamental difference between segmenting by time versus ensemble. Ultimately the full ensemble space is covered with either partitioning approach.
- Shuffle the order of neuron position's to see effect on resulting LZ dictionary.
- Investigate repeated presentation of neural firing signal.
- Extend the general approach to account for noise in neural firings.
- Update the complexity normalization technique to account for piecewise presentation. Richer insights may be available from analyzing the structure of the dictionary tree such as how balanced it is as well as path lengths rather than only looking at its size.

5. ALGORITHM DEVELOPMENT FROM NEUROGENESIS THEORIES

The ultimate goal of our project was the development of novel algorithmic strategies to address potential mission area challenges using the observations made from studying the neurogenesis process. We pursued two areas as potential application impacts of neurogenesis, the use of new neurons in making existing machine learning model adaptive over time, and the use of a neurogenesis-like process to facilitate compressive sensing.

5.1. Continuous adaptation of models

While machine learning and ANNs utilize neural terminology and allude to concepts such as learning, in reality most of these methods lack substantial capabilities to update models continuously or online; rather these methods typically involve the generation of a data-driven model based on a set of (typically labeled) data at the onset, train using that data, and then deploy in a fixed condition. If the statistics of the relevant world changes; the originally-developed model loses some degree of utility. Once conditions have evolved substantially, usually the necessary investment in a new model framework is made.

5.1.1. Model evolution formalism

Suppose a model, M_A exists and is based on data about some condition in the world W_A acquired at T=0. Notably, this model is actually based on observations of the world, D_A, which is, of course, a limited view of what the world actually is. This model, M_A, is designed such that if a new set of data, d(t), arrives, the operator can make some prediction *p* based on the learned data D_A and implicitly the underlying world W_A.

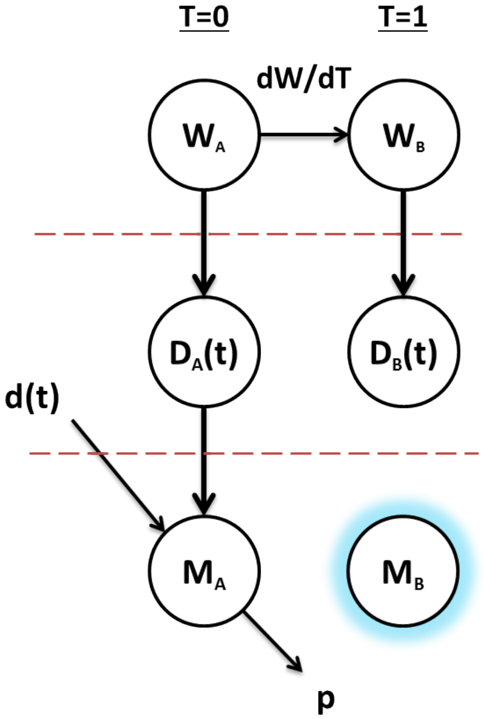


Figure 41 - Relevance of data-driven models after world changes considerably

Our challenge is what to do if the world changes. In particular, how should our model M be updated when we know that the world has changed at some rate over time? In the case of a machine learning classifier, such as deep learning, this could be as simple as the objects under investigation experiencing context drift (e.g., the appearance of planes has changed somewhat over the last 100 years); or it could be as sophisticated as a major shift in how agents in the world interact with each other in a model of global dynamics.

5.1.2. Abstract example of neurogenesis-inspired online plasticity algorithm

Here, we will review a sampling of several key types of neural plasticity that touch on several different time and influence scales. We will describe these mechanisms here in the context of a simple linear separation model, which can be thought of as a very simple approximation of a neuron.

$$f(x) = \begin{cases} 1 \text{ if } a_1x_1 + a_2x_2 > b \\ 0 \quad\quad\quad \text{otherwise} \end{cases} \tag{49}$$

In this direction, a primary research goal would be to determine how best to generalize these techniques to more complex mission relevant models.

The dominant form of continuous adaptation of neural system involves the dynamic alteration of the strength of the connections between neurons (i,e., synaptic weights). In our simple linear model example, the synaptic weights would relate to a_1 and a_2. Suppose that after the model is initially generated (Figure 41a), new information arises from an SME or other source for which the original model is no longer adequate. While in this trivial case one could simply recompute the ideal discriminator (a technique that does not necessarily work in more complex models) the synaptic plasticity method would dynamically update the weighting parameters (e.g., lowering a_1, increasing a_2) to refit the model continuously (Figure 41b).

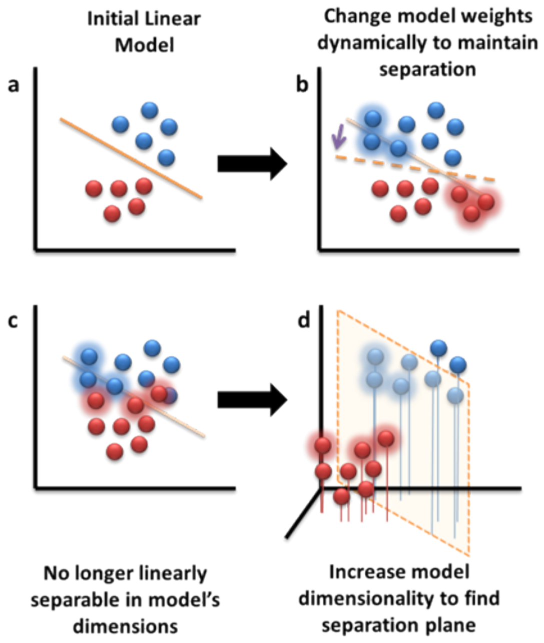

Initial Linear Model
Change model weights dynamically to maintain separation
a
b
c
d
No longer linearly separable in model's dimensions
Increase model dimensionality to find separation plane

Figure 42 - Illustration of neural plasticity based algorithm to impact classification

One concern with updating conventional models in this manner would be the attribution of error due to novel data across all possible parameters (this relates to the vanishing gradient problem in traditional backpropagation learning), particularly in a dynamic setting where the value of the new data is unknown. We foresee three brain-inspired modifications alleviating this concern in our work. First, unlike most models, neural synaptic weights are multi-dimensional; effectively some synapses are more sensitive than others to new information than others, and this sensitivity is itself a state that can vary over time. This relates to directly to trust; if a single datum from a suboptimal source arises that challenges as trusted parameter, a key parameter should not necessarily drift. Second, we anticipate that neural reinforcement learning mechanisms can be used to retroactively harden, or anneal, those parameters whose recent changes produced a desirable behavior. Finally, in order to keep the models from overfitting to new data, we plan to leverage sparse optimization techniques to direct dynamics of the model to the fewest number of parameters that produce the desired outcome, which is consistent with how the brain distributes information across its synaptic parameters.

A bigger challenge arises in both real neural systems and conventional models when new information is presented that makes the previous model inadequate. Figure 41c shows an example where the linear separation is no longer possible in the two dimensions on which the data is projected. For such data, techniques such as the kernel trick employed by support vector machines (SVMs) are often effective at allowing discrimination in a higher dimensional space. Alternatively, the brain uses techniques such as synaptogenesis (creating new connections) and neurogenesis (creating new nodes) to address this problem. Suppose a third dimension to the data exists that previously was unnecessary for the discriminator, but as new data arises, it becomes useful in forming the discrimination. In this example, the generation of a new parameter or connection in the model (considering x_3 in addition to x_1 and x_2) allows a separation plane to be formed that returns the model to an effective state (Figure 41d).

Beyond a few specific algorithms built around the concept (such as Adaptive Resonance Theory [ART] networks), the concept of structural plasticity is not widely considered in machine learning. Nonetheless, there is growing evidence that this is one of the most potent mechanisms the brain uses for adapting itself over time, particularly in response to substantial shifts in data representations . Clearly, this is a much more challenging technique to integrate in a model in order to preserve stability, so the major research challenges faced will involve not only developing methods to implement it in other model types, but also identifying which model structures permit this form of plasticity and, related, whether and features of the original data may indicate that this form of plasticity will be required.

5.2. Compressed Sensing

Detection of familiar faces or interesting objects in images or video is a task that humans can do better (faster and more accurately) than computers . More generally processing of sensory input in the context of memory formation and recall is managed much more efficiently in biological systems. With respect to images, we can and do build cameras with increasing levels of resolution. These devices do indeed produce visually appealing images, but are they really an efficient use of technology? A technique called compressed sensing has been developed to reduce the amount of raw data necessary in producing an accurate image . In fact compressed sensing has been used to develop a one-pixel camera and compared with mega-pixel images from typical commercially available cameras . These results demonstrate the use of more efficient image construction using random sampling below the Nyquist rate, specifically in cases where the images being reconstructed are sparse or compressible. Ganguli and Sompolinsky describe how compressed sensing might be used in the brain to facilitate both compression and expansion of neural firing patterns . In this document we will describe how the unique structure and location (both physically and functionally) of the hippocampus and its components can facilitate compressed sensing during episodic memory formation and recall. We will also describe a neural-inspired algorithm to address the problem of efficient sensory data fusion, in a "big data" setting.

5.2.1. Assumptions

In this section we will describe several key assumptions we are using throughout this document. First, we assume a major function of the hippocampus is memory formation and recall. The kind of memory we are interested in in this paper is that which involves the hippocampus, which can be either episodic or spatial memory formation and recall. In this paper, we will not address other types of memory or memory systems in the brain, such as procedural memory. Another assumption we will use in this paper is that fact that the hippocampus receives input from other cortex through the entorhinal cortex (EC) and then returns its output back to the EC. This specific neural formation from EC through the hippocampus, generally to the DG then through regions of the cornu Ammonis (CA), at least CA3, and finally either directly or through the subiculum back to the EC again, forms a loop, partially shown in Figure 40.

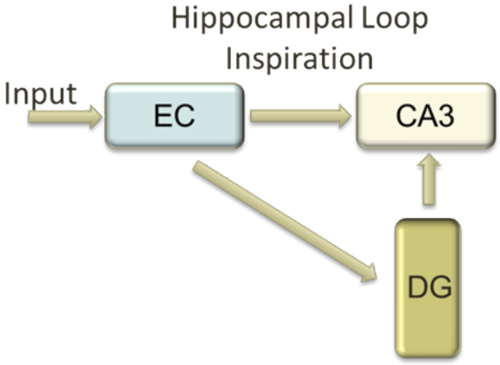


Figure 43 - Simple flowchart of neural processing pathways in the hippocampus.

We assume that both during memory formation and recall many passes through the hippocampal loop are used. We also assume that active neurogenesis in the DG is integral to at least memory recall. Another (more complete) view of the hippocampal loop is shown in Figure 41. In this figure the recall loop involves (potentially) all of cortex that provides input to the EC as well as that which receives output from the EC.

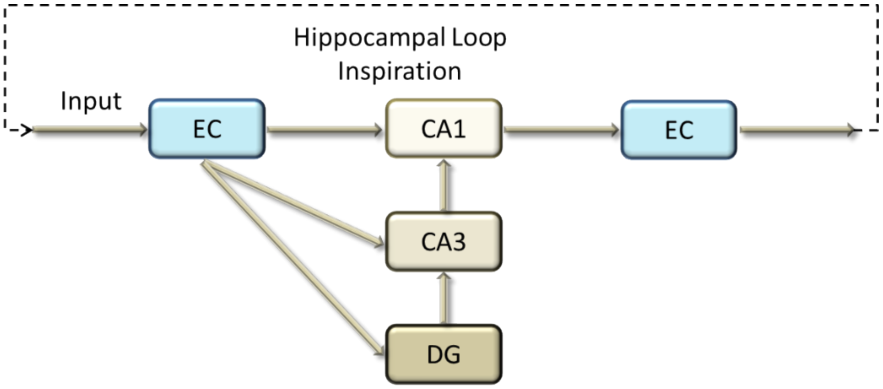


Figure 44 - Simple flowchart of neural processing pathways involving the hippocampus and surrounding cortex.

The hippocampal loop in Figure 41includes other systems and functions, such as consolidation of short-term to long-term memory .

5.2.2. Memory Formation and Reconstruction Using the Hippocampus

A framework for using both the compressed coding and randomized reconstruction aspects of compressed sensing for long range communication in the brain has been previously described . In this section we will describe how the location, structure and function of the hippocampus are ideal for facilitating compressed sensing.

The hippocampus has several functions. One function of the hippocampus is to form episodic memories , and another is in the formation of spatial memory {Moser, 2008 #53}. The hippocampus is also involved in recall of these types of memories. The connections from entorhinal cortex into the dentate gyrus correspond to compression. Entorhinal cortex receives input from many areas of the brain, which are highly distributed and non-sparse. In the formation of memories, activation in the dentate gyrus is sparse, such that these patterns of activation can represent specific events in an episodic sequence. We will describe in this paper how the hippocampus with neurogenesis facilitates recall of memories with compressed sensing.

With respect to the recall of memory through the hippocampus structure, think of the process looping through the hippocampus possibly many times. The recall itself as well as the saliency of the recalled memory can be initiated outside of the hippocampus, but looping through the hippocampus allows very highly distributed neural signals from many areas of the brain to be focused (fused/aggregated) together using the sparse representation of the dentate gyrus. Very specific sparse combinations of the mature granule cells of the dentate gyrus act as keys to specific memory events, keyed to other more distributed areas of the brain where specific details (sensory activations, thoughts, emotions) are manifest. Looping through the hippocampus allows this seemingly disparate activity to be brought together in one area. Later recall of specific events can be easily managed if the neural keys are still in the dentate gyrus, but even if they are no longer around similar keys can be formed allowing access to these memories. The description so far has not mentioned compressed sensing, but that will be discussed next.

As described so far (above), the memory recall would necessitate compressed sensing, but it could require a vast number of neural states and memory keys in the dentate gyrus to represent the many, many possible memories encountered in everyday life. One way to facilitate more efficient memory encoding and thus recall is to use a mechanism such as compressed sensing. In this situation very detailed but much lower resolution components/parts of a memory are distributed throughout the brain (outside the hippocampus), and then random sampling, a la compressed sensing, is used to reconstruct/recombine the memory into a whole. The hippocampus, specifically the dentate gyrus, is an ideal brain structure for facilitating this random sampling, and since it (the DG) is also part of the looping mechanism for memory construction (described above) it fits as an integral part. And neurogenesis is what seeds this random sampling. New neurons (GCs) in the dentate gyrus initially form connections along existing pathways, which means that these new neurons form at least partial links with existing memory keys, but they also can facilitate random sampling with respect to these memory keys. Immature dentate gyrus granule cell neurons are in general more active than mature cells. This hyperactivity of the immature neurons can be thought of as adding randomness or noise to the process of memory recall, which facilitates compressed sensing.

5.2.3. Using Neurogenesis to Randomly Sample during Memory Reconstruction

Figure 3 shows our general conceptual idea for using compressed sensing inspired by the neural architecture and function of the hippocampus in creating more efficient high fidelity aggregate sensors from smaller local (distributed) low fidelity sensor elements.

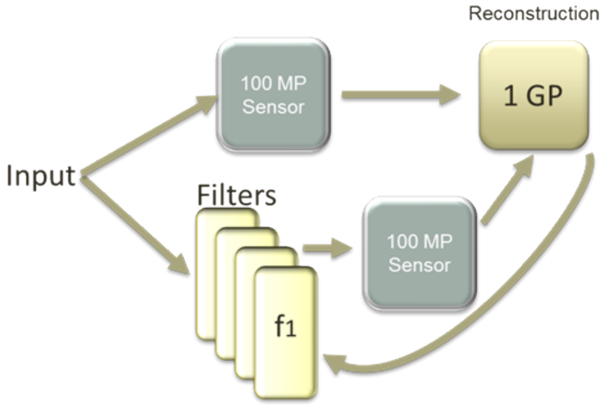


Figure 45 - Conceptual diagram for reconstructing memory using aggregate sensor filters and compressed sensing technology.

In the diagram in Figure 42, the ‘filters’ represent separate passes through the memory recall loop (in the hippocampus) described above (see Figure 40), where each filter represents an at least slightly different random projection of the low fidelity sensor data being aggregated into the high fidelity reconstruction. With this conceptual model, we propose to develop domain specific filters for processing sensor data (most likely an image) in a neural-inspired reconstruction loop. Thus, we will produce a higher resolution sensor map (image) from lower resolution physical sensor (camera).

Another view of compressed sensing using the hippocampal loop is shown in Figure 43. In this figure instead of filters in the hippocampus we have keys to specific memory elements. These elements are concentrated upon smaller regions of cortex and/or sensory input, and thus they are lower in resolution than the desired overall reconstructed memory and most likely overlapping.

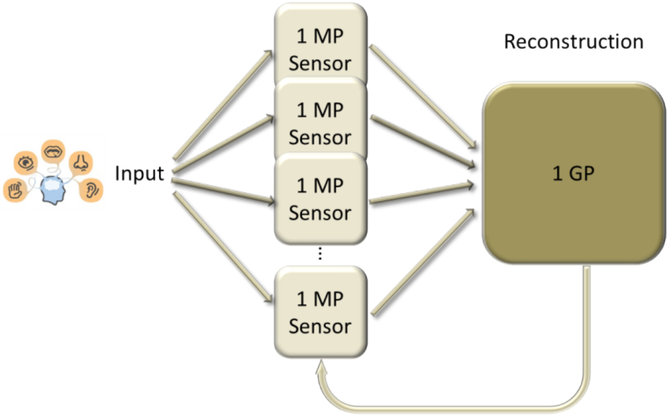


Figure 46 - Conceptual diagram for reconstructing memory using distributed aggregate sensor component and compressed sensing technology.

In this version of our conceptual model, the DG neurons represent keys to memory elements from cortex (accessed through EC) instead of filters. For mature DG neurons these keys are specifically linked to a particular memory event. Young DG neurons, which are only partially connected to one (or more) mature neurons, facilitate randomized projections into the keyed memory space (beyond EC). These younger neurons then serve as a randomized walk in memory space during reconstruction of a memory (via multiple passes through the reconstruction loop) using compressed sensing technology. Figure 44 shows a specific example of image reconstruction using this conceptual model.

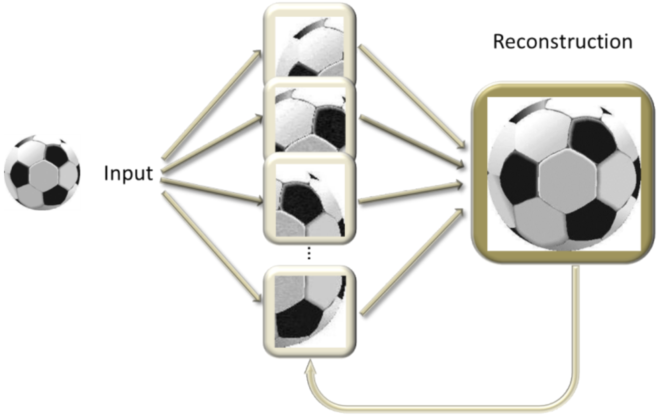


Figure 47 - Example reconstruction of "soccer ball" image using conceptual model.

In general EC inputs to the hippocampus already include completed objects, and the function of the hippocampus is to put these objects together in context with space and time sequences which are keyed by the DG neurons. We are showing this example because it is easy to grasp, and we believe straightforward to implement. We propose to use (overlapping) distributed low resolution sensors (cameras) in a neural-inspired compressed sensing (loop) algorithm for generating high resolution images. Future extensions to this conceptual model could include object fusion in more complex scene contexts.

5.2.4. Conclusions/Future Directions

In this paper, we have described a neural-inspired framework for facilitating compressed sensing for memory formation and recall using variations of neural processing loop described by the architecture of the hippocampus. Both conceptual models will need to be extended to include other (non-image) sensory data. In this setting the reconstructed memory element will also have more dimensions (rather than the 2 dimensions shown conceptually in Figure 42 and Figure 43 above).

6. CONCLUSIONS

Although neurogenesis has been extensively studied by the neuroscience community, the details of its computational impact on both cognition and potential algorithmic utility have remained uncertain. While there have been a number of theories substantiated in part by behavioral experimentation, most of these studies have only indirectly related neurogenesis to potential functions. The high fidelity modeling described here is notable in large part because it is the first study of neurogenesis to study the effects of new neurons at realistic scales. For a number of reasons, scaling neural systems is non-trivial; the non-linearities of neurons and the sparse representation of information in distributed ensembles raises challenges when drawing inferences from reduced scale models. In the case of neurogenesis, where the relative activity of young neurons and mature neurons is in large part due to the effective scale difference between the different populations, reduced scale models greatly reduce the marginal impact of young neurons.

Our observed impact of young neurons in preferentially encoding novel information that is not represented by mature neurons has substantial potential algorithmic impact. Conventional machine learning approaches have long suffered from the stability-plasticity dilemma, which precludes systems trained on one set of information from effectively learning a new set of information without risking either forgetting the previously trained data or failing to encode the new data. Neurogenesis provides a potentially unique method for getting around this challenge, though future work will be required to fully understand its algorithmic potential.

In conclusion, this work illustrates the value of bringing substantial computational resources towards understanding the function of a neural system. Further, it demonstrates a path by which neural insights can be converted into potentially novel algorithmic solutions to address mission critical challenges.

7. REFERENCES

Aimone, J. B., W. Deng, and F. H. Gage. "Adult Neurogenesis: Integrating Theories and Separating Functions." [In eng]. *Trends Cogn Sci* 14, no. 7 (Jul 2010): 325-37.

———. "Resolving New Memories: A Critical Look at the Dentate Gyrus, Adult Neurogenesis, and Pattern Separation." [In eng]. *Neuron* 70, no. 4 (May 26 2011): 589-96.

Aimone, J. B., and F. H. Gage. "Modeling New Neuron Function: A History of Using Computational Neuroscience to Study Adult Neurogenesis." [In eng]. *Eur J Neurosci* 33, no. 6 (Mar 2011): 1160-9.

Aimone, J. B., J. Wiles, and F. H. Gage. "Computational Influence of Adult Neurogenesis on Memory Encoding." [In eng]. *Neuron* 61, no. 2 (Jan 29 2009): 187-202.

Aimone, James B, Wei Deng, and Fred H Gage. "Adult Neurogenesis in the Dentate Gyrus." In *Space, Time and Memory in the Hippocampal Formation*. 409-29: Springer, 2014.

Aimone, James B, Yan Li, Star W Lee, Gregory D Clemenson, Wei Deng, and Fred H Gage. "Regulation and Function of Adult Neurogenesis: From Genes to Cognition." *Physiological Reviews* 94, no. 4 (2014): 991-1026.

Bakker, A., C. B. Kirwan, M. Miller, and C. E. Stark. "Pattern Separation in the Human Hippocampal Ca3 and Dentate Gyrus." [In eng]. *Science* 319, no. 5870 (Mar 21 2008): 1640-2.

Baraniuk, Richard G. "Single-Pixel Imaging Via Compressive Sampling." *IEEE Signal Processing Magazine* (2008).

Candes, Emmanuel J, Justin K Romberg, and Terence Tao. "Stable Signal Recovery from Incomplete and Inaccurate Measurements." *Communications on pure and applied mathematics* 59, no. 8 (2006): 1207-23.

Chambers, R Andrew, and Susan K Conroy. "Network Modeling of Adult Neurogenesis: Shifting Rates of Neuronal Turnover Optimally Gears Network Learning According to Novelty Gradient." *Journal of cognitive neuroscience* 19, no. 1 (2007): 1-12.

Chambers, R Andrew, Marc N Potenza, Ralph E Hoffman, and Willard Miranker. "Simulated Apoptosis/Neurogenesis Regulates Learning and Memory Capabilities of Adaptive Neural Networks." *Neuropsychopharmacology: official publication of the American College of Neuropsychopharmacology* 29, no. 4 (2004): 747-58.

Clelland, C. D., M. Choi, C. Romberg, G. D. Clemenson, Jr., A. Fragniere, P. Tyers, S. Jessberger*, et al.* "A Functional Role for Adult Hippocampal Neurogenesis in Spatial Pattern Separation." [In eng]. *Science* 325, no. 5937 (Jul 10 2009): 210-3.

Clopath, Claudia, and Wulfram Gerstner. "Voltage and Spike Timing Interact in Stdp–a Unified Model." *Frontiers in synaptic neuroscience* 2 (2010).

Cohen, NJ, and H Eichenbaum. "Memory, Amnesia, and the Hippocampal System, 1993." *Cambridge, MA: MIT Press* 3 (378-89.

Cover, Thomas M, and Joy A Thomas. "Elements of Information Theory 2nd Edition." (2006).

Crick, Christopher, and Willard Miranker. "Apoptosis, Neurogenesis, and Information Content in Hebbian Networks." *Biological cybernetics* 94, no. 1 (2006): 9-19.

Deisseroth, Karl, Sheela Singla, Hiroki Toda, Michelle Monje, Theo D Palmer, and Robert C Malenka. "Excitation-Neurogenesis Coupling in Adult Neural Stem/Progenitor Cells." *Neuron* 42, no. 4 (2004): 535-52.

Dieni, Cristina V, Angela K Nietz, Roberto Panichi, Jacques I Wadiche, and Linda Overstreet-Wadiche. "Distinct Determinants of Sparse Activation During Granule Cell Maturation." *The Journal of Neuroscience* 33, no. 49 (2013): 19131-42.

Donoho, David L. "Compressed Sensing." *Information Theory, IEEE Transactions on* 52, no. 4 (2006): 1289-306.

Eriksson, P. S., E. Perfilieva, T. Bjork-Eriksson, A. M. Alborn, C. Nordborg, D. A. Peterson, and F. H. Gage. "Neurogenesis in the Adult Human Hippocampus." [In eng]. *Nat Med* 4, no. 11 (Nov 1998): 1313-7.

Esposito, M. S., V. C. Piatti, D. A. Laplagne, N. A. Morgenstern, C. C. Ferrari, F. J. Pitossi, and A. F. Schinder. "Neuronal Differentiation in the Adult Hippocampus Recapitulates Embryonic Development." *J Neurosci* 25, no. 44 (Nov 2 2005): 10074-86.

Ganguli, Surya, and Haim Sompolinsky. "Compressed Sensing, Sparsity, and Dimensionality in Neuronal Information Processing and Data Analysis." *Annual review of neuroscience* 35 (2012): 485-508.

Ge, S., C. H. Yang, K. S. Hsu, G. L. Ming, and H. Song. "A Critical Period for Enhanced Synaptic Plasticity in Newly Generated Neurons of the Adult Brain." [In eng]. *Neuron* 54, no. 4 (May 24 2007): 559-66.

Gu, Y., M. Arruda-Carvalho, J. Wang, S. R. Janoschka, S. A. Josselyn, P. W. Frankland, and S. Ge. "Optical Controlling Reveals Time-Dependent Roles for Adult-Born Dentate Granule Cells." [In eng]. *Nat Neurosci* 15, no. 12 (Nov 2012): 1700-6.

Hafting, T., M. Fyhn, S. Molden, M. B. Moser, and E. I. Moser. "Microstructure of a Spatial Map in the Entorhinal Cortex." [In eng]. *Nature* 436, no. 7052 (Aug 11 2005): 801-6.

Hancock, Peter JB, Vicki Bruce, and A Mike Burton. "Recognition of Unfamiliar Faces." *Trends Cogn Sci* 4, no. 9 (2000): 330-37.

Hargreaves, E. L., G. Rao, I. Lee, and J. J. Knierim. "Major Dissociation between Medial and Lateral Entorhinal Input to Dorsal Hippocampus." [In eng]. *Science* 308, no. 5729 (Jun 17 2005): 1792-4.

Indiveri, Giacomo, Bernabé Linares-Barranco, Tara Julia Hamilton, André Van Schaik, Ralph Etienne-Cummings, Tobi Delbruck, Shih-Chii Liu*, et al.* "Neuromorphic Silicon Neuron Circuits." *Frontiers in neuroscience* 5 (2011).

Insel, Thomas R, Story C Landis, and Francis S Collins. "The Nih Brain Initiative." *Science* 340, no. 6133 (2013): 687-88.

Isely, Guy, Christopher Hillar, and Fritz Sommer. "Deciphering Subsampled Data: Adaptive Compressive Sampling as a Principle of Brain Communication." Paper presented at the Advances in neural information processing systems, 2010.

Izhikevich, Eugene M. "Simple Model of Spiking Neurons." *Neural Networks, IEEE Transactions on* 14, no. 6 (2003): 1569-72.

Kanerva, Pentti. *Sparse Distributed Memory*. MIT press, 1988.

Lempel, Abraham, and Jacob Ziv. "On the Complexity of Finite Sequences." *Information Theory, IEEE Transactions on* 22, no. 1 (1976): 75-81.

Li, Y., J. B. Aimone, X. Xu, E. M. Callaway, and F. H. Gage. "Development of Gabaergic Inputs Controls the Contribution of Maturing Neurons to the Adult Hippocampal Network." [In eng]. *Proc Natl Acad Sci U S A* 109, no. 11 (Mar 13 2012): 4290-5.

Marin-Burgin, A., L. A. Mongiat, M. B. Pardi, and A. F. Schinder. "Unique Processing During a Period of High Excitation/Inhibition Balance in Adult-Born Neurons." [In eng]. *Science* 335, no. 6073 (Mar 9 2012): 1238-42.

McNaughton, B. L., and R.G.M. Morris. "Hippocampal Synaptic Enhancement and Information Storage within a Distributed Memory System." *Trends Neurosci.* 10, no. 10 (1987).

Merolla, Paul A, John V Arthur, Rodrigo Alvarez-Icaza, Andrew S Cassidy, Jun Sawada, Filipp Akopyan, Bryan L Jackson*, et al.* "A Million Spiking-Neuron Integrated Circuit with a Scalable Communication Network and Interface." *Science* 345, no. 6197 (2014): 668-73.

Mongiat, L. A., M. S. Esposito, G. Lombardi, and A. F. Schinder. "Reliable Activation of Immature Neurons in the Adult Hippocampus." [In eng]. *PLoS One* 4, no. 4 (2009): e5320.

Mu, Y., C. Zhao, and F. H. Gage. "Dopaminergic Modulation of Cortical Inputs During Maturation of Adult-Born Dentate Granule Cells." [In eng]. *J Neurosci* 31, no. 11 (Mar 16 2011): 4113-23.

O'Reilly, R. C., and J. L. McClelland. "Hippocampal Conjunctive Encoding, Storage, and Recall: Avoiding a Trade-Off." [In eng]. *Hippocampus* 4, no. 6 (Dec 1994): 661-82.

Pouille, Frédéric, Antonia Marin-Burgin, Hillel Adesnik, Bassam V Atallah, and Massimo Scanziani. "Input Normalization by Global Feedforward Inhibition Expands Cortical Dynamic Range." *Nat Neurosci* 12, no. 12 (2009): 1577-85.

Sahay, A., and R. Hen. "Adult Hippocampal Neurogenesis in Depression." [In eng]. *Nat Neurosci* 10, no. 9 (Sep 2007): 1110-5.

Sahay, A., D. A. Wilson, and R. Hen. "Pattern Separation: A Common Function for New Neurons in Hippocampus and Olfactory Bulb." [In eng]. *Neuron* 70, no. 4 (May 26 2011): 582-8.

Santarelli, L., M. Saxe, C. Gross, A. Surget, F. Battaglia, S. Dulawa, N. Weisstaub*, et al.* "Requirement of Hippocampal Neurogenesis for the Behavioral Effects of Antidepressants." [In eng]. *Science* 301, no. 5634 (Aug 8 2003): 805-9.

Solstad, Trygve, Edvard I Moser, and Gaute T Einevoll. "From Grid Cells to Place Cells: A Mathematical Model." *Hippocampus* 16, no. 12 (2006): 1026-31.

Spalding, K. L., O. Bergmann, K. Alkass, S. Bernard, M. Salehpour, H. B. Huttner, E. Bostrom*, et al.* "Dynamics of Hippocampal Neurogenesis in Adult Humans." [In eng]. *Cell* 153, no. 6 (Jun 6 2013): 1219-27.

Squire, LR, and DL Schacter. "Neuropsychology of Memory. New York." Guilford Press, 2002.

Szczepanski, Janusz, José M Amigó, E Wajnryb, and MV Sanchez-Vives. "Application of Lempel-Ziv Complexity to the Analysis of Neural Discharges." *Network: Computation in Neural Systems* 14, no. 2 (2003): 335-50.

Tank, DavidW, and JOHNJ HOPFIELD. "Collective Computation in Neuronlike Circuits." *Scientific American* 257, no. 6 (1987): 104-14.

Toni, N., D. A. Laplagne, C. Zhao, G. Lombardi, C. E. Ribak, F. H. Gage, and A. F. Schinder. "Neurons Born in the Adult Dentate Gyrus Form Functional Synapses with Target Cells." [In eng]. *Nat Neurosci* 11, no. 8 (Aug 2008): 901-7.

Treves, A., and E. T. Rolls. "Computational Constraints Suggest the Need for Two Distinct Input Systems to the Hippocampal Ca3 Network." [In eng]. *Hippocampus* 2, no. 2 (Apr 1992): 189-99.

Treves, A., A. Tashiro, M. P. Witter, and E. I. Moser. "What Is the Mammalian Dentate Gyrus Good For?" [In eng]. *Neuroscience* 154, no. 4 (Jul 17 2008): 1155-72.

Victor, Jonathan D. "Approaches to Information-Theoretic Analysis of Neural Activity." *Biological theory* 1, no. 3 (2006): 302.

Vineyard, Craig M, Stephen J Verzi, Thomas P Caudell, Michael L Bernard, and James B Aimone. "Adult Neurogenesis: Implications on Human and Computational Decision Making." In *Foundations of Augmented Cognition*. 531-40: Springer, 2013.

Wiskott, L., M. J. Rasch, and G. Kempermann. "A Functional Hypothesis for Adult Hippocampal Neurogenesis: Avoidance of Catastrophic Interference in the Dentate Gyrus." [In eng]. *Hippocampus* 16, no. 3 (2006): 329-43.

Yassa, M. A., A. T. Mattfeld, S. M. Stark, and C. E. Stark. "Age-Related Memory Deficits Linked to Circuit-Specific Disruptions in the Hippocampus." [In eng]. *Proc Natl Acad Sci U S A* 108, no. 21 (May 24 2011): 8873-8.

Yassa, M. A., and C. E. Stark. "Pattern Separation in the Hippocampus." [In eng]. *Trends Neurosci* 34, no. 10 (Oct 2011): 515-25.

Zhao, C., E. M. Teng, R. G. Summers, Jr., G. L. Ming, and F. H. Gage. "Distinct Morphological Stages of Dentate Granule Neuron Maturation in the Adult Mouse Hippocampus." [In eng]. *J Neurosci* 26, no. 1 (Jan 4 2006): 3-11.

Ziv, Jacob, and Abraham Lempel. "Compression of Individual Sequences Via Variable-Rate Coding." *Information Theory, IEEE Transactions on* 24, no. 5 (1978): 530-36.

———. "A Universal Algorithm for Sequential Data Compression." *IEEE Transactions on information theory* 23, no. 3 (1977): 337-43.

DISTRIBUTION

1	MS0359	D. Chavez, LDRD Office	1911
1	MS0899	Technical Library	9536 (electronic copy)

Sandia National Laboratories